LOCALIZED GRAPHLET CORRELATION FOR EVALUATING GRAPH GENERATIVE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Evaluating graph generative models requires understanding which structural patterns a model captures and which it fails to reproduce. To gain such insight, models are typically evaluated on their ability to generate graphs structurally similar to a reference set of procedurally generated graphs with known properties, using hand-crafted structural descriptors to quantify the similarity. Empirically, the most discriminative graph descriptor is the average orbit count vector, the mean, over all nodes, of how often each node participates in each automorphism orbit of small non-isomorphic induced subgraphs known as graphlets. However, averaging discards how orbit counts co-vary across nodes, and computing correlations globally fails to account for local structural relationships, such as whether nodes co-occur in the same graphlet instance. We introduce *localized graphlet correlation*, a family of 20 structural descriptors that condition pairwise orbit count correlations on orbit adjacency, a matrix encoding how often two nodes co-occur in the same graphlet instance, with each node at a specific orbit. Each feature is interpretable as an assortativity coefficient for a pair of orbit positions. Applied to three synthetic benchmark datasets and four generative models, localized graphlet correlation is the most discriminative descriptor overall, and reveals that current models can preserve marginal orbit count distributions while disrupting local co-variation: failures undetectable using existing descriptors.

1 INTRODUCTION

Graph generative models are increasingly essential for advancing fields that rely on structured relational data, enabling the generation of realistic synthetic graphs (You et al., 2018) and the exploration of novel designs in applications such as molecular drug discovery (Wang et al., 2023b), protein engineering (Xinhui et al., 2024), and urban infrastructure optimization (Wettewa et al., 2024). To gain insight into which structural patterns a model captures and which it fails to reproduce, graph generative models are typically evaluated on their ability to generate graphs structurally similar to a reference set of procedurally generated graphs with known properties, using hand-crafted structural descriptors to quantify the similarity (Vignac et al., 2022; Chen et al., 2025; Liao et al., 2019; Bergmeister et al., 2024; Krimmel et al., 2026).

To describe graph structure, state-of-the-art evaluation pipelines rely on hand-crafted graph statistics (see Appendix H), as they are directly interpretable and do not require task-specific training. Empirically, the best-performing graph descriptor for evaluating graph generative models is the *average orbit count* vector (Krimmel et al., 2026), which averages, over all nodes, how often each node participates in each (automorphism) orbit of small induced non-isomorphic subgraphs, known as *graphlets* (Pržulj, 2007). This is consistent with theoretical results showing that substructure counts are strictly more expressive than the Weisfeiler-Lehman test (Wang et al., 2023a), which bounds the expressiveness of Graph Neural Networks (GNNs) (see Figure 4B for an illustration of the nine up to four node graphlets and their fifteen orbits). However, averaging orbit counts across nodes into a single vector discards how counts of different orbit types co-vary across nodes. To address this, the *Graphlet Correlation Matrix* (GCM) describes a graph through pairwise *graphlet correlations*—the Spearman rank correlation between how often nodes participates in two orbit types (Yaveroğlu et al., 2014). Despite this theoretical advantage, the GCM has not been benchmarked against other descriptors for evaluating graph generative models.

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

However, since graphlet correlations are computed globally across all nodes, without conditioning on their connectivity relationships (e.g., whether nodes are connected or not, or how they are arranged within local neighborhoods), two networks can exhibit identical graphlet correlations while differing substantially in their connectivity structure. Together, these limitations imply that current graph descriptors often lack sufficient discriminative power to distinguish generated from reference graphs, leading to ambiguous model rankings, obscuring failure modes, and ultimately constraining progress towards robust, application-ready graph generative systems.

To address the lack of topological conditioning in the GCM, we first observe that the Spearman rank correlation between two orbit count vectors can be expressed as a normalized Euclidean inner product, which motivates the following topology-aware generalization of graphlet correlation.

- We introduce *localized graphlet correlation*, which generalizes standard graphlet correlation by replacing the Euclidean inner product with one induced by orbit adjacency matrices (Windels et al., 2026), encoding how often pairs of nodes co-occur within specific graphlet orbits (Section 2.5).
- Since orbit adjacency matrices are neither Hermitian nor positive definite, we construct valid inner product spaces via the Haar adjacency for directed graphs (Badea & Dumitrescu, 2025) and spectral filtering (Section 2.5).
- We show that the resulting descriptor is interpretable: it is analogous to an assortativity coefficient, measuring whether nodes with high counts of one orbit type tend to be connected, via orbit adjacency, to nodes with high counts of another (Section 2.4).

We benchmark localized graphlet correlation against existing descriptors on their ability to detect structural discrepancies between generated and reference graphs, across four generative models and three synthetic datasets. To compare descriptor quality, we use PolyGraph Discrepancy (PGD) (Krimmel et al., 2026), whose intrinsic scale allows comparison across descriptors, unlike the widely used Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) (Section 2.6). Localized graphlet correlation (Assort.) is the most discriminative descriptor overall, ranking first for 6 out of 12 model-dataset combinations and first or second for 9 out of 12, with a mean improvement of 14 percentage points over the non-localized GCM, benchmarked here for the first time in this setting (Section 3.1; see Figure 1 for an illustration). On SBM-L, all existing descriptors assign near-zero scores for AutoGraph and DiGress, while Assort. is the only descriptor that finds signal, outperforming the next best descriptor by up to 4.7×. Perturbation experiments (Section 3.2) confirm why: localized correlations detect disruptions to orbit co-occurrence structure even when marginal orbit count distributions are preserved, precisely the failure mode exhibited by these models. The computational overhead is comparable to existing descriptors and negligible relative to model training (Section 3.3).

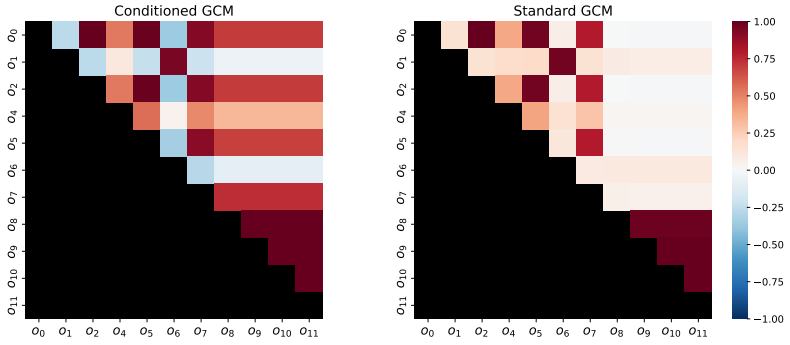


Figure 1: Conditioning on orbit adjacency uncovers local structural correlations not captured by AutoGraph. Left: average conditioned GCM (conditioned on orbit adjacency for orbits 1 and 2) for AutoGraph-generated LOBSTER-L graphs that PolyGraph detects as generated with confidence ≥ 0.95. Right: standard GCM for these same graphs. Trained on the standard GCM, PolyGraph confidence drops to 0.80 on average for these same graphs, illustrating that conditioning reveals structural differences that matter to distinguish generated from reference graphs, invisible to the standard GCM.

2 METHODS

In this section, we introduce our localized graphlet correlation metric, which generalizes the standard graphlet correlation by weighting the correlation of orbit counts according to the local structural relationships between nodes, as captured by orbit adjacencies. We first show that the standard graphlet correlation can be interpreted as a dot product in Euclidean space (Section 2.1), making it straightforward to generalize by replacing the Euclidean inner product with one that encodes local network topology. Since orbit adjacency matrices are in general neither Hermitian nor positive definite — both properties required for a valid (strict) inner product space — we proceed in two steps: we first define a Hermitian variant based on the orbit adjacency (Section 2.2), then apply spectral filtering to obtain a positive semi-definite product space (Section 2.3). The resulting localized graphlet correlation is analogous to degree assortativity, measuring the tendency of nodes with high counts of orbit o_i to be connected, via a given orbit adjacency, to nodes with high counts of orbit o_j (Section 2.4). A summary of mathematical notation is provided in Appendix A. Illustrations of graphlets, orbits, and the orbit adjacency matrix are in Appendix C. The 28 orbit adjacencies are illustrated in Appendix D.

2.1 GRAPHLET CORRELATION AS AN INNER PRODUCT

The graphlet correlation between orbit count vectors o_i and o_j is the Spearman correlation $\rho_s(o_i, o_j)$ (Yaveroğlu et al., 2014) (see Appendix H). To generalize it to take local graph structure into account, we first express it as a cosine similarity in a Euclidean inner product space. To do so, we use the fact that Spearman correlation equals Pearson correlation on ranks (Spearman, 1961). To write Pearson correlation as a cosine similarity, we mean-center the rank vectors.

Formally, for a graph with n nodes, let $r(\cdot)$ denote the elementwise (mid-)rank transform and $\tilde{o}_i := r(o_i) - \frac{1}{n}\mathbf{1}^\top r(o_i)\mathbf{1}$ the mean-centered rank (where $\mathbf{1}$ is the vector of all ones) vector of orbit o_i:

$$\begin{aligned}\rho_s(o_i, o_j) &= \frac{\operatorname{cov}(r(o_i),\, r(o_j))}{\sqrt{\operatorname{cov}(r(o_i),\, r(o_i))}\,\sqrt{\operatorname{cov}(r(o_j),\, r(o_j))}} \\ &= \frac{\frac{1}{n}\tilde{o}_i^\top \tilde{o}_j}{\sqrt{\frac{1}{n}\tilde{o}_i^\top \tilde{o}_i}\,\sqrt{\frac{1}{n}\tilde{o}_j^\top \tilde{o}_j}} = \frac{\tilde{o}_i^\top I\, \tilde{o}_j}{\sqrt{\tilde{o}_i^\top I\, \tilde{o}_i}\,\sqrt{\tilde{o}_j^\top I\, \tilde{o}_j}},\end{aligned} \tag{1}$$

where the second equality uses $\operatorname{cov}(r(o_i), r(o_j)) = \frac{1}{n}\tilde{o}_i^\top \tilde{o}_j$ since $\tilde{o}_i$ is mean-centered by construction; the third equality cancels the common $\frac{1}{n}$ factors; and I is the identity matrix.

Since I weights every node equally, Equation 1 correlates orbit counts uniformly across all node pairs, without regard to the structural relationships between nodes. Two graphs can therefore exhibit identical graphlet correlations while differing substantially in how orbit counts are organised relative to the orbit adjacency structure. To address this, we replace I in Equation 1 with orbit-adjacency-based inner product matrices, defined in the following sections.

2.2 HERMITIAN MATRICES BASED ON ORBIT ADJACENCY

For a matrix $M \in \mathbb{R}^{n\times n}$ or $\mathbb{C}^{n\times n}$ to define a valid inner product space via $\langle x, y\rangle_M = x^* M y$, there are two conditions:

1. Symmetry (or Hermitian in the complex case): for all vectors $x, y \in \mathbb{R}^{n\times 1}$ or $\mathbb{C}^{n\times 1}$, it holds that $\langle x, y\rangle_M = \overline{\langle y, x\rangle_M}$.
2. Positive definiteness: for all non-zero vectors $x \in \mathbb{R}^{n\times 1}$ or $\mathbb{C}^{n\times 1}$, it holds that $x^* M x > 0$.

Orbit adjacency matrices $A_{o_a\text{-}o_b}$ are in general neither symmetric nor positive definite, and therefore do not define a valid inner product space. However, by interpreting each orbit adjacency matrix as the adjacency matrix of a weighted, directed graph, where the weight of an edge from node u to node v is given by the entry $A_{o_a\text{-}o_b}(u, v)$, we can apply spectral methods for directed graphs to construct valid inner product spaces. We adopt the Haar adjacency from the Haar Laplacian framework for weighted, directed graphs (Badea & Dumitrescu, 2025) since it preserves directionality (unlike symmetrization),

requires no hyperparameter tuning (unlike the magnetic Laplacian (He et al., 2022; Zhang et al., 2021)), and produces distinct adjacency matrices for anti-parallel edges (unlike the sign-magnetic Laplacian (Fiorini et al., 2023)). Combining Haar-adjacency with orbit adjacency, we define the *Haar-orbit adjacency* matrix, $H_{o_a\text{-}o_b}$, as:

$$H_{o_a\text{-}o_b} = \frac{1}{2}\big(A_{o_a\text{-}o_b} + A^{\top}_{o_a\text{-}o_b}\big) + \frac{\imath}{2}\big(A_{o_a\text{-}o_b} - A^{\top}_{o_a\text{-}o_b}\big)\,, \tag{2}$$

where $\imath = \sqrt{-1}$ denotes the imaginary unit. The real part of H_{uv} captures symmetric co-occurrence; the imaginary part captures directional asymmetry.

Let $D_{o_a\text{-}o_b}$ be the diagonal degree matrix defined from $H_{o_a\text{-}o_b}$ as:

$$D_{o_a\text{-}o_b} = \operatorname{diag}\big(|H_{o_a\text{-}o_b}|\,\mathbf{1}\big), \tag{3}$$

where $|H_{o_a\text{-}o_b}|$ denotes the element-wise absolute value of $H_{o_a\text{-}o_b}$ and $\mathbf{1}$ is the vector of all ones. We define the *normalized Haar-orbit adjacency* as:

$$\tilde{H}_{o_a\text{-}o_b} = D^{-1/2}_{o_a\text{-}o_b}\, H_{o_a\text{-}o_b}\, D^{-1/2}_{o_a\text{-}o_b}, \tag{4}$$

By construction, $\tilde{H}_{o_a\text{-}o_b}$ inherits the properties of the normalized Haar adjacency (Badea & Dumitrescu, 2025): it is Hermitian, has real eigenvalues, and its eigenvectors form a unitary matrix. However, it is not necessarily positive definite.

2.3 Positive semi-definite inner product spaces via spectral filtering

While $\tilde{H}_{o_a\text{-}o_b}$ is Hermitian, it is not necessarily positive definite, and therefore does not define a valid inner product space. For the Cauchy–Schwarz inequality to hold, necessary for $\mathrm{Re}(\rho_{loc}), \mathrm{Im}(\rho_{loc}) \in [-1, 1]$ (see Appendix I), it suffices that the bilinear form be induced by a positive semi-definite matrix. Concretely, to capture the leading cluster structure of the orbit adjacency (von Luxburg, 2007), we skip the largest eigenvalue (whose eigenvector is constant for a fully connected graph), retain the eigenvectors corresponding to the k largest remaining eigenvalues clipped to ≥ 0, and set all other eigenvalues to zero, yielding a positive semi-definite matrix.

While $\tilde{H}_{o_a\text{-}o_b}$ is Hermitian, it is not necessarily positive definite, and therefore does not define a valid inner product space. For the Cauchy–Schwarz inequality to hold — necessary for $\mathrm{Re}(\rho_{loc}), \mathrm{Im}(\rho_{loc}) \in [-1, 1]$ (see Appendix I) — it suffices that the bilinear form be induced by a positive semi-definite matrix. We achieve this by applying a spectral filter to $\tilde{H}_{o_a\text{-}o_b}$ that also isolates the leading cluster structure of the orbit adjacency (von Luxburg, 2007): we discard the largest eigenvalue (whose eigenvector is constant for a fully connected graph), clip the next k largest eigenvalues to ≥ 0, and set all remaining eigenvalues to zero.

Formally, let $U\Lambda U^*$ be the eigendecomposition of $\tilde{H}_{o_a\text{-}o_b}$, with eigenvalues ordered by decreasing magnitude. We define the filter

$$f(\lambda_i) = \begin{cases} 0, & \text{if } i = 1 \\ \max(\lambda_i, 0), & \text{if } 2 \leq i \leq k+1 \\ 0, & \text{otherwise} \end{cases}, \tag{5}$$

and set $\hat{H}_{o_a\text{-}o_b} = U f(\Lambda) U^*$. Since all entries of $f(\Lambda)$ are non-negative, $\hat{H}_{o_a\text{-}o_b}$ is Hermitian and positive semi-definite, defining a valid semi-inner product $\langle x, y\rangle = x^{\top} \hat{H}_{o_a\text{-}o_b}\, y$ on $\mathbb{R}^n$. The *localized graphlet correlation* of orbit counts o_i and o_j, conditioned on orbit adjacency $\hat{H}_{o_a\text{-}o_b}$, is then defined as:

$$\rho_{loc}(o_i, o_j \mid \hat{H}_{o_a\text{-}o_b}) = \frac{\tilde{o}_i^{\top} \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j}{\sqrt{\tilde{o}_i^{\top} \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_i}\, \sqrt{\tilde{o}_j^{\top} \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j}}, \tag{6}$$

where the orbit count vectors are real, so $\tilde{o}_i^* = \tilde{o}_i^{\top}$. Replacing $\hat{H}_{o_a\text{-}o_b}$ with I recovers the standard graphlet correlation of Equation 1. Numerical edge cases and computational considerations are discussed in Appendix G.

2.4 INTERPRETATION OF LOCALIZED GRAPHLET CORRELATIONS AS 'ASSORTATIVITY' COEFFICIENTS

The localized graphlet correlation $\rho_{loc}(o_i, o_j \mid \hat{H}_{o_a\text{-}o_b})$ measures whether nodes with high counts of orbit o_i tend to be connected, via $A_{o_a\text{-}o_b}$, to nodes with high counts of orbit o_j. Substituting $\tilde{H}_{o_a\text{-}o_b} = D_{o_a\text{-}o_b}^{-1/2} H_{o_a\text{-}o_b} D_{o_a\text{-}o_b}^{-1/2}$, the bilinear form expands as:

$$\tilde{o}_i^\top \tilde{H}_{o_a\text{-}o_b}\, \tilde{o}_j = \sum_{u,v} \frac{H_{o_a\text{-}o_b}(u,v)}{\sqrt{D_{o_a\text{-}o_b}(u,u)\, D_{o_a\text{-}o_b}(v,v)}}\, \tilde{o}_i(u)\, \tilde{o}_j(v). \tag{7}$$

The spectral filter (Section 2.3) retains the k dominant spectral components of this sum, and ρ_{loc} (Equation 6) normalizes the filtered version to $[-1, 1]$.

Equation 7 is analogous to Newman's assortativity (Newman, 2002; 2003), which measures whether nodes tend to connect to nodes with similar scalar labels, weighted by the plain adjacency. Our formulation differs from Newman's in three ways: (1) graph edges are replaced by orbit adjacency $A_{o_a\text{-}o_b}$, capturing graphlet co-occurrence beyond edges; (2) edge weights are degree-normalized, preventing high-degree nodes from dominating; and (3) scalar node labels are replaced by rank-transformed vectors.

2.5 LOCALIZED GRAPHLET CORRELATION AS A GRAPH DESCRIPTOR

To describe the structure of a graph, the GCM (see Appendix H) computes pairwise graphlet correlations over non-redundant orbit pairs for up to four-node graphlets (**GCM4**), yielding $\binom{11}{2} = 55$ features. Setting $\hat{M} = I$ in Equation 6 recovers each GCM4 correlation. Replacing $\hat{M} = I$ with $\hat{H}_{o_a\text{-}o_b}$ conditions each correlation on the structural relationship between orbit types (o_a, o_b), yielding a *conditioned GCM* that captures how orbit counts co-vary locally, weighted by graphlet co-occurrence structure, rather than uniformly across all nodes. Each conditioned GCM is evaluated independently as a separate descriptor; we select the most discriminative one empirically via PolyGraph (Section 3).

We consider the 27 orbit adjacencies $A_{o_a\text{-}o_b}$ for orbit pairs at most 2 hops apart in four-node graphlets. Of these, 16 are asymmetric and form 8 transpose pairs: since the Haar matrix of $A_{o_b\text{-}o_a}$ is the conjugate of that of $A_{o_a\text{-}o_b}$, the resulting conditioned GCMs carry identical information, reducing the 27 to 19 unique orbit-adjacency-based conditioned GCMs. Including $\hat{M} = I$, which recovers GCM4 as a special case, yields **20 GCMs** in total (19 conditioned plus the standard GCM). For symmetric orbit adjacencies ($a = b$), each yields 55 real-valued features; for Hermitian adjacencies ($a \neq b$), the 55 complex-valued features are flattened to 110 real-valued features (real and imaginary parts) for compatibility with PolyGraph. In view of its interpretation as a graphlet assortativity coefficient (Section 2.4), we refer to localized graphlet correlation as **Assort.** hereafter.

2.6 EVALUATION PROTOCOL

Evaluating graph generative models requires: (1) a graph descriptor summarizing the structure of the graphs in the reference and generated set, and (2) a similarity or distance metric to compare the descriptor values for the graphs in the reference and generated set. The most widely used evaluation metric for graph generative models is Maximum Mean Discrepancy (MMD) (Gretton et al., 2012), which quantifies the divergence between the distributions of structural descriptors computed on reference and generated graphs. Despite its popularity, MMD has several limitations (O'Bray et al., 2021): it is highly sensitive to hyperparameter choices and lacks an intrinsic scale, making scores difficult to compare across descriptors, models, or graph domains.

PolyGraph Discrepancy (PGD) addresses these issues by training a TabPFN classifier (Hollmann et al., 2022) to distinguish reference from generated graphs based on a given descriptor, thereby yielding an estimate on the Jensen–Shannon (JS) distance between the corresponding distributions of generated and reference graphs (Krimmel et al., 2026). PGD is bounded in $[0, 1]$, enabling consistent comparison across descriptors, models, and graph types. PGD is part of the PolyGraph evaluation framework, which provides benchmark datasets comprising synthetic and real-world graphs across diverse domains and generative model paradigms, including autoregressive models (GRAN (Liao et al., 2019), AutoGraph (Chen et al., 2025)) and diffusion models (ESGG (Bergmeister et al., 2024), DiGress (Vignac et al., 2022)).

Because PGD operates on descriptors rather than directly on graphs, it provides a lower bound on the true JS distance, with tighter bounds obtained by more expressive descriptors. When the reference and generated distributions differ, PGD scores can therefore be used to evaluate descriptors: for a given generative model and reference set, a higher PGD indicates a more discriminative descriptor.

3 EXPERIMENTS

We present three experiments. Section 3.1 benchmarks localized graphlet correlation (Assort.) against existing descriptors on three synthetic datasets and four generative models, showing it is the most discriminative descriptor overall. Section 3.2 uses controlled perturbation experiments to explain this advantage: correlation-based descriptors are sensitive to topology-preserving perturbations that leave orbit count distributions unchanged, exposing a failure mode invisible to existing descriptors. Section 3.3 shows that the computational cost of Assort. is comparable to the slowest existing descriptors.

3.1 BENCHMARK EVALUATION

To avoid inflating performance through repeated testing, we evaluate Assort. as a single descriptor by flattening and concatenating conditioned GCMs and the standard (unconditioned) GCM into a single feature vector per graph. Doing so naively, concatenating all 19 conditioned GCMs, requires 19 eigendecompositions. To avoid these costly computations, we perform an ablation study (Appendix E) and show that it suffices to use the unconditioned GCM and the 5 conditioned GCMs based on the generating orbit adjacencies from which all remaining adjacencies are derived, reducing eigendecompositions from 19 to 5. Further ablation of the remaining design choices leads us to recommend top-k spectral filtering, degree-normalized orbit adjacencies, rank-transformed orbit counts, and Haar symmetrization. All results in the main text use this configuration.

We evaluate four graph generative models (AutoGraph, DiGress, ESGG, GRAN) on three benchmark datasets (LOBSTER-L, PLANAR-L, SBM-L; see Appendix B), following the PolyGraph evaluation protocol (Krimmel et al., 2026). For each descriptor, PGD is estimated by drawing 10 bootstrap samples of 2,048 graphs uniformly at random from both the reference test set (4,096 graphs) and the generated set. Each sample is evaluated via stratified 4-fold cross-validation, where TabPFN is trained to discriminate reference from generated graphs and the fold score is the Jensen-Shannon divergence estimated from the classifier's predicted probabilities. The final PGD is the mean over the 10 bootstrap scores; we report mean $\pm$ standard deviation in Table 1. We set $k = \lfloor\sqrt{n/2}\rfloor$ following (von Luxburg, 2007); Appendix K shows results are robust to this choice: Assort. remains the best-performing subscore in 6 out of 12 model–dataset combinations across all values of k. Appendix J verifies calibration: PGD scores are close to zero when distinguishing two subsets of the same reference distribution.

Assort. is the most discriminative descriptor overall. It ranks first for 6 out of 12 model–dataset combinations and first or second for 9 out of 12. The nearest competitor is GCM5 (to the best of our knowledge, not previously benchmarked for this application), which ranks first for 3 out of 12 combinations and first or second for 5 out of 12. All three cases where GCM5 leads occur on PLANAR-L, with Assort. a close second: 52.8 vs. 62.7 (AutoGraph), 99.3 vs. 99.5 (GRAN), and 56.0 vs. 56.0 (ESGG, tied).

Since GCM4 is the special case of our method with $\hat{M} = I$, the gap between GCM4 and Assort. directly quantifies the contribution of the orbit adjacency inner product space. Assort. matches or outperforms GCM4 on all 12 combinations, with a mean improvement of 14 points and gaps ranging from 2 to 27 points, confirming that the locality of the inner product space, not merely the use of correlations, drives the discriminative gains. We analyse the computational cost of this gain in expressiveness in Section 3.3.

The signal uncovered by Assort. on SBM-L is striking: for AutoGraph and DiGress, it outperforms the second-best descriptor by 19.6 points (27.0 vs. GIN=7.4) and 19.8 points (36.2 vs. Orb5=16.4) respectively, with all other descriptors scoring near zero. This margin indicates that Assort. is recovering structural signal entirely invisible to existing descriptors on this dataset. A feature importance analysis (Appendix F) shows that the GCM conditioned on $A_{o_{14}\text{-}o_{14}}$ accounts for over 50% of Assort.'s PGD score on SBM-L. Since these models match marginal orbit counts (Orbit4=3.7,

Table 1: PGD scores ($\times 100$) per descriptor, generative model, and dataset (mean $\pm$ std over 10 subsamples). For each model–dataset combination, the most discriminative descriptor (highest PGD) is shown in **bold** and the second-most discriminative is underlined. The quality of a generative model is measured by its maximum PGD across all descriptors, which gives the tightest available lower bound on the Jensen–Shannon distance between the reference and generated distributions. Per dataset, the model with the lowest maximum PGD (hardest to distinguish from the reference) is indicated by a **bold** row label; the second-lowest by an underlined row label. All scores are multiplied by 100 for legibility.

Dataset	Model	PGD (↓)	PGD subscores (↑)								
			Assort.	GCM4	GCM5	Orb4.	Orb5.	Deg.	Eig.	Clust.	GIN
PLANAR-L	AutoGraph	71.1 ± 1.0	57.6 ± 1.1	28.7 ± 2.0	**71.1** ± 1.0	31.6 ± 1.0	34.3 ± 1.4	8.8 ± 1.6	28.7 ± 2.0	9.6 ± 2.4	15.7 ± 2.1
	DIGRESS	46.6 ± 1.8	43.0 ± 1.9	26.8 ± 1.7	44.7 ± 1.4	43.0 ± 1.3	**46.7** ± 1.7	25.5 ± 1.9	39.9 ± 1.9	28.5 ± 2.2	29.7 ± 1.7
	GRAN	99.4 ± 0.6	99.2 ± 1.1	99.3 ± 1.0	**99.5** ± 0.5	98.8 ± 1.0	99.4 ± 0.9	98.2 ± 0.7	98.9 ± 1.1	99.3 ± 0.6	92.6 ± 0.7
	ESGG	66.4 ± 0.8	60.7 ± 1.0	38.9 ± 1.5	**66.4** ± 0.8	45.1 ± 1.6	47.7 ± 1.1	26.4 ± 1.2	31.0 ± 1.7	14.6 ± 3.7	36.8 ± 2.2
LOBSTER-L	AutoGraph	90.5 ± 0.8	**90.5** ± 0.8	66.6 ± 1.1	70.2 ± 0.9	15.0 ± 1.8	16.6 ± 1.4	11.0 ± 1.5	11.2 ± 2.0	5.0 ± 0.8	13.1 ± 1.5
	DIGRESS	95.3 ± 0.7	**95.3** ± 0.7	77.7 ± 1.4	79.5 ± 1.3	5.7 ± 1.7	5.6 ± 1.8	1.7 ± 1.7	0.8 ± 1.7	2.5 ± 0.6	3.2 ± 2.1
	GRAN	88.2 ± 0.7	**88.3** ± 0.8	87.4 ± 0.7	87.7 ± 0.6	85.6 ± 0.7	86.0 ± 0.6	78.2 ± 0.6	70.7 ± 1.1	21.3 ± 0.6	83.9 ± 0.8
	ESGG	88.4 ± 0.6	**88.4** ± 0.6	82.3 ± 0.8	86.0 ± 0.7	66.0 ± 0.7	70.6 ± 0.6	64.1 ± 1.1	56.0 ± 1.3	0.0 ± 0.0	69.7 ± 1.2
SBM-L	**AutoGraph**	24.2 ± 1.4	**24.2** ± 1.4	2.9 ± 1.9	1.9 ± 2.1	3.7 ± 2.9	3.9 ± 2.3	5.3 ± 2.0	2.8 ± 2.2	1.4 ± 1.3	6.8 ± 1.0
	DIGRESS	33.3 ± 1.0	**33.3** ± 1.0	9.0 ± 2.7	17.2 ± 2.1	14.8 ± 1.5	16.8 ± 1.8	10.3 ± 1.8	7.6 ± 3.1	7.3 ± 2.1	13.9 ± 1.5
	GRAN	68.0 ± 1.2	64.8 ± 1.0	56.9 ± 1.0	63.6 ± 1.1	64.1 ± 1.2	**68.0** ± 1.2	58.3 ± 1.9	59.5 ± 1.6	50.5 ± 1.8	63.8 ± 1.1
	ESGG	99.3 ± 0.2	94.7 ± 0.5	92.2 ± 0.4	96.5 ± 0.5	90.3 ± 0.6	95.8 ± 0.3	97.3 ± 0.5	**99.3** ± 0.2	97.1 ± 0.4	97.9 ± 0.3

Orbit5=4.1 for AutoGraph), this reveals a specific failure mode: these models generate graphs whose marginal orbit count distributions closely match the reference, but whose local co-occurrence structure does not. We hypothesize this reflects a shared architectural constraint. In DiGress, the reverse sampling step draws all edges independently from their predicted marginals, preventing joint enforcement of co-occurrence constraints. In AutoGraph, the autoregressive generation order means edges to not-yet-visited nodes are invisible at prediction time, creating blind spots for co-occurrence patterns that span multiple generation steps. In both cases, orbit marginals can be matched implicitly through learned statistics, while conditional co-occurrence requires reasoning about jointly-determined edges that neither architecture directly enforces.

3.2 DESCRIPTOR SENSITIVITY TO STRUCTURAL PERTURBATIONS

In Table 1, GCM4 outperforms Orbit4 in 6 out of 12 combinations, illustrating that correlations between orbit counts contribute to discriminatory power. Conversely, Orbit4 outperforms GCM4 in the remaining 6 combinations, and in 2 of these — PLANAR-L/DiGress and SBM-L/GRAN — it outperforms even the localized Assort. To understand what drives these observations, we measure PGD for Orbit4, GCM4, and Assort. at increasing noise levels p, comparing 2,048 reference graphs against 2,048 perturbed graphs across four perturbation types. For the edge rewiring perturbations, preserving total edge count, and the edge swapping perturbations, preserving node degrees, we report values at the noise level where the gap between correlation-based and count-based descriptors is largest, directly quantifying the maximum sensitivity advantage.

- **Edge deletion**: remove a fraction p of edges uniformly at random, decreasing edge density.
- **Edge addition**: add a fraction p of new edges uniformly at random, increasing edge density.
- **Edge rewiring**: for a fraction p of edges, remove edge (u, v) and add a new edge (u, w) for a randomly chosen w, preserving edge count.
- **Edge swapping**: for a fraction p of edges, pick two edges (u, v) and (x, y) and replace with (u, x) and (v, y), preserving the degree sequence exactly.

Figure 2 shows results for LOBSTER-L. For edge deletion and addition, Orbit4 is more sensitive than GCM4 and Assort.: orbit counts capture absolute edge frequencies and are directly sensitive to density changes, while rank-based correlations are invariant to such rescaling and lose this signal. The key differences emerge under edge rewiring and edge swapping perturbations. For edge rewiring ($p = 0.05$), GCM4 and Assort. substantially outperform Orbit4: PGD of 0.70 and 0.71 versus 0.47. For edge swapping ($p = 0.26$), the advantage is smaller but still present: GCM4=0.77 and Assort=0.83 versus Orbit4=0.69. To confirm that this advantage stems from local co-occurrence structure rather than marginal count differences, we compute the Jensen–Shannon divergence between

per-node orbit count histograms of reference and perturbed graphs. On SBM-L/AutoGraph, the mean divergence across all 11 orbits is 0.0001; marginals are effectively identical, yet Assort. achieves a PGD of 24.2 (Table 1), confirming that the discriminative signal captured by correlation-based descriptors is invisible to count-based ones.

For SBM-L (Figure 7c in the supplement), the same pattern holds but is weaker: for edge rewiring ($p = 0.11$), GCM4=0.89, Assort=0.91, Orbit4=0.88; for edge swapping ($p = 0.11$), GCM4=0.84, Assort=0.86, Orbit4=0.82. The smaller gap reflects the fact that Orbit4 is already highly sensitive to rewiring on SBM-L: rewiring disrupts community structure in ways that directly affect marginal orbit count distributions, so the correlation-based advantage is reduced. On LOBSTER-L, rewiring breaks orbit co-occurrence patterns without substantially changing marginal orbit counts, which is precisely where correlation-based descriptors gain their advantage. For PLANAR-L (Figure 7a in the supplement), differential sensitivity between descriptors is not observable: at the smallest perturbation level ($p \approx 0.05$), almost all descriptors immediately saturate near PGD $= 1$, so the relative ordering is uninformative.

Taken together, these results help explain the benchmark pattern: when a model's failure mode disrupts orbit co-occurrence structure while preserving marginal orbit count distributions, as AutoGraph and DiGress do on LOBSTER-L, GCM4 and its localized extension Assort. are far more discriminative than Orbit4. When the failure mode alters orbit count distributions directly, Orbit4 can match or exceed correlation-based descriptors, as for DiGress on PLANAR-L.

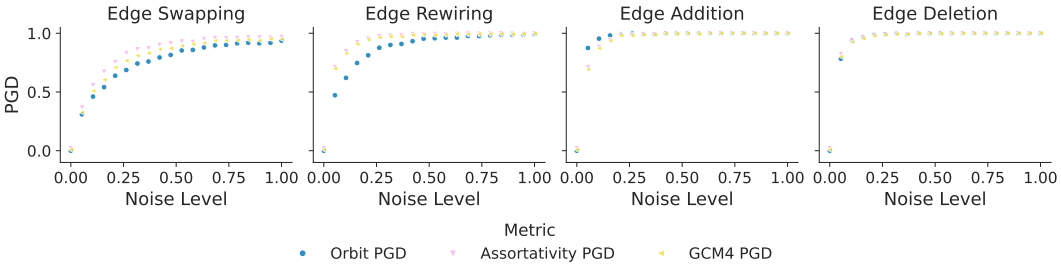


Figure 2: PGD as a function of noise level p for Orbit4, GCM4, and Assort. on LOBSTER-L across four perturbation types.

3.3 DISCRIMINATORY POWER VS. COMPUTATIONAL COST

A key practical question is whether the increased discriminatory power of localized graphlet correlation justifies its computational cost. Figure 3 shows each descriptor's PGD subscore against its median per-graph computation time on AutoGraph, the model whose outputs most closely resemble the reference distribution on two out of the three datasets (Table 1). All timings are measured on a single CPU node (Intel Xeon Silver 4116 @ 2.10 GHz, 8 cores, 8 GB RAM). GIN is included in the figure for completeness but, as a neural network, would typically run on GPU; we exclude it from the timing discussion. Results for AutoGraph are shown in Figure 3; results for the remaining models are in Appendix M. For Assort., the time shown is for the slowest of the 20 conditioned GCMs computed in parallel.

Among CPU-bound descriptors, Assort. is the slowest per graph, yet the absolute cost is modest: under 9 s for the entire reference test set (4,096 graphs) on LOBSTER-L and PLANAR-L, under 15 s on SBM-L. On SBM-L, where Assort. matters most, it is only 1.5× slower than GCM5, the next slowest CPU-bound descriptor. Even computed sequentially (a factor of 20, one per conditioned GCM), the full test set completes in under 5 minutes, negligible compared to the hours required to train a generative model.

Whether this cost is justified depends on the model being evaluated. For models that already score high on simpler descriptors, such as GRAN and ESGG on LOBSTER-L, the additional cost of Assort. yields only marginal gains. For models whose outputs are harder to distinguish from the reference, such as AutoGraph and DiGress on SBM-L, Assort. is the only descriptor that finds meaningful signal: scores of 27.0 and 36.2, versus the second-best CPU-bound descriptor Deg=5.7 and Orb5=16.4 respectively (a 4.7× and 2.2× improvement).

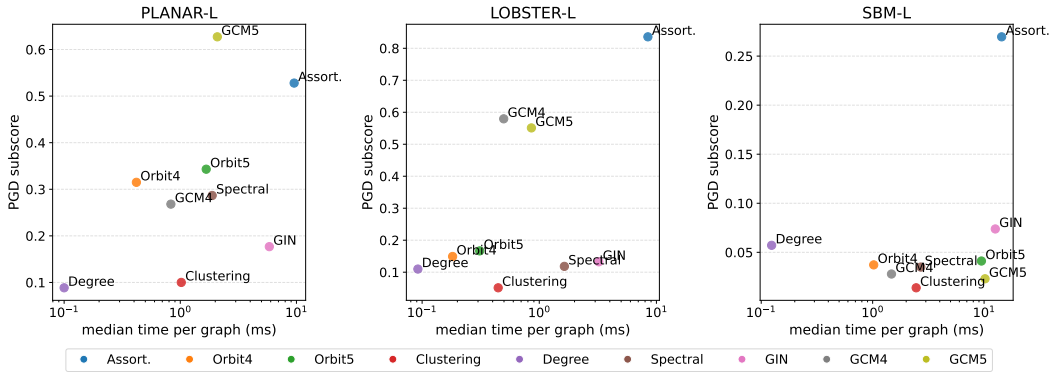


Figure 3: Per-descriptor discriminatory power (PGD subscore on AutoGraph) vs. median per-graph computation time (ms, over reference graphs). Top row: PLANAR-L, LOBSTER-L; bottom row: SBM-L.

4 CONCLUSION

We introduced localized graphlet correlation, a family of structural descriptors that weight orbit count correlations by local graph structure. Unlike neural descriptors, each feature is interpretable as an assortativity coefficient for a specific pair of orbit positions (Section 2.4), giving direct insight into which structural patterns a model fails to reproduce. Localizing correlations reveals a class of failure modes invisible to existing descriptors: generative models that match marginal orbit count distributions but fail to reproduce how orbits co-occur conditionally on local connectivity. On SBM-L, this is the dominant failure mode for AutoGraph and DiGress, where all existing descriptors assign near-zero discriminative scores while localized graphlet correlation is the only descriptor that finds signal (Table 1). Perturbation experiments confirm this mechanistically: correlation-based descriptors are selectively sensitive to perturbations that leave marginal orbit counts unchanged (Section 3.2). The computational overhead is comparable to existing descriptors and negligible compared to model training, and is most justified precisely where it matters most: for models whose outputs are hardest to distinguish from the reference (Section 3.3).

5 LIMITATIONS AND FUTURE WORK

Scalability. Computing localized graphlet correlation has two complexity bottlenecks. The first is orbit adjacency counting (Section 2.1), performed by GRADCO (Windels et al., 2026) with time complexity $O(no^2i)$, where o and i are the maximum out- and in-degree in the degree-directed graph representation of G. GRADCO is marginally slower in theory than ORCA (Hočevar & Demšar, 2014) (used by Orbit4, $O(no^3)$) because it additionally enumerates diamond subgraphs; diamonds are rare in sparse graphs, however, and empirical runtimes are comparable. The second bottleneck is the eigendecomposition for spectral filtering (Section 2.3), with complexity $O(n^3)$, the same as the existing Spectral descriptor. The method will therefore scale to whatever graph sizes Orbit4 and Spectral support. In practice, current graph generative models are mostly evaluated on small to medium-sized graphs, so neither bottleneck poses an issue for the current application. Sampling-based approximations for orbit counting (Bourreau et al., 2024) are a natural direction to push this further.

Extension to larger graphlets. The current method localizes correlations between four-node orbit counts using four-node orbit adjacency matrices as the spectral filter. A natural extension is to localize GCM5 using the same four-node orbit adjacencies, without requiring enumeration of the five-node orbit adjacencies. Extending orbit adjacency itself to five-node graphlets is non-trivial.

Scope. All experiments are conducted on synthetic, unlabeled graphs. The framework is not specific to orbit counts: any real-valued node label vector can be substituted as the signal to localise, enabling a straightforward extension to attributed graphs and other domains.

Reproducibility statement

All code and data will be made publicly available upon publication.

AI use statement

In this work, we used generative AI tools (Claude, Anthropic) to integrate the localized graphlet correlation method into the PolyGraph evaluation pipeline and to polish the manuscript draft. We have not used generative AI tools to formulate or implement the method, prove theorems, generate data, or interpret results. All AI-assisted code and text was reviewed by the authors. We take responsibility for the final content of this work.

References

Theodor-Adrian Badea and Bogdan Dumitrescu. Haar-Laplacian for Directed Graphs. *IEEE Transactions on Signal and Information Processing over Networks*, 11:1238–1253, 2025. ISSN 2373-776X, 2373-7778. doi: 10.1109/TSIPN.2025.3611242. URL https://ieeexplore.ieee.org/document/11168271/.

Andreas Bergmeister, Karolis Martinkus, Nathanaël Perraudin, and Roger Wattenhofer. Efficient and scalable graph generation through iterative local expansion. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=2XkTz7gdpc.

Yann Bourreau, Marco Bressan, T-H. Hubert Chan, Qipeng Kuang, and Mauro Sozio. Efficient streaming algorithms for graphlet sampling. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 20081–20110. Curran Associates, Inc., 2024. doi: 10.52202/079017-0634. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/23d64d26abb5a0e9f2014cfcc700f82a-Paper-Conference.pdf.

Dexiong Chen, Markus Krimmel, and Karsten Borgwardt. Flatten Graphs as Sequences: Transformers are Scalable Graph Generators. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, October 2025. URL https://openreview.net/forum?id=eszmES7j1F.

Stefano Fiorini, Stefano Coniglio, Michele Ciavotta, and Enza Messina. SigMaNet: One Laplacian to Rule Them All. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):7568–7576, June 2023. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v37i6.25919. URL https://ojs.aaai.org/index.php/AAAI/article/view/25919.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. ISSN 1533-7928. URL http://jmlr.org/papers/v13/gretton12a.html.

Yixuan He, Michael Perlmutter, Gesine Reinert, and Mihai Cucuringu. MSGNN: A Spectral Graph Neural Network Based on a Novel Magnetic Signed Laplacian. In *Proceedings of the First Learning on Graphs Conference*, pp. 40:1–40:39. PMLR, December 2022. URL https://proceedings.mlr.press/v198/he22c.html.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *NeurIPS 2022 First Table Representation Workshop*, 2022. URL https://openreview.net/forum?id=eu9fVjVasr4.

Tomaž Hočevar and Janez Demšar. A combinatorial approach to graphlet counting. *Bioinformatics*, 30(4):559–565, 02 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btt717. URL https://doi.org/10.1093/bioinformatics/btt717.

Markus Krimmel, Philip Hartout, Karsten Borgwardt, and Dexiong Chen. Polygraphscore: a classifier-based metric for evaluating graph generative models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nGizKVBCgn.

Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, Will Hamilton, David K Duvenaud, Raquel Urtasun, and Richard Zemel. Efficient graph generation with graph recurrent attention networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/d0921d442ee91b896ad95059d13df618-Paper.pdf.

Karolis Martinkus, Andreas Loukas, Nathanaël Perraudin, and Roger Wattenhofer. SPECTRE: Spectral Conditioning Helps to Overcome the Expressivity Limits of One-shot Graph Generators. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 15159–15179. PMLR, June 2022. URL https://proceedings.mlr.press/v162/martinkus22a.html.

M. E. J. Newman. Assortative Mixing in Networks. *Physical Review Letters*, 89(20):208701, October 2002. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.89.208701. URL https://link.aps.org/doi/10.1103/PhysRevLett.89.208701.

M. E. J. Newman. Mixing patterns in networks. *Physical Review E*, 67(2):026126, February 2003. ISSN 1063-651X, 1095-3787. doi: 10.1103/PhysRevE.67.026126. URL https://link.aps.org/doi/10.1103/PhysRevE.67.026126.

Leslie O'Bray, Max Horn, Bastian Rieck, and Karsten Borgwardt. Evaluation Metrics for Graph Generative Models: Problems, Pitfalls, and Practical Solutions. In *International Conference on Learning Representations*, October 2021. URL https://openreview.net/forum?id=tBtoZYKd9n.

Nataša Pržulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23 (2):e177–e183, 01 2007. ISSN 1367-4803. doi: 10.1093/bioinformatics/btl301. URL https://doi.org/10.1093/bioinformatics/btl301.

Charles Spearman. The proof and measurement of association between two things. 1961.

Rylee Thompson, Boris Knyazev, Elahe Ghalebi, Jungtaek Kim, and Graham W. Taylor. On evaluation metrics for graph generative models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=EnwCZixjSh.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. DiGress: Discrete Denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, September 2022. URL https://openreview.net/forum?id=UaAD-Nu86WX.

Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, December 2007. ISSN 1573-1375. doi: 10.1007/s11222-007-9033-z. URL https://doi.org/10.1007/s11222-007-9033-z.

Qing Wang, Dillon Ze Chen, Asiri Wijesinghe, Shouheng Li, and Muhammad Farhan. $\mathscr{N}$-WL: A new hierarchy of expressivity for graph neural networks. In *The Eleventh International Conference on Learning Representations*, 2023a. URL https://openreview.net/forum?id=5cAI0qXxyv.

Yuyang Wang, Zijie Li, and Amir Barati Farimani. Graph Neural Networks for Molecules. In Chen Qu and Hanchao Liu (eds.), *Machine Learning in Molecular Sciences*, pp. 21–66. Springer International Publishing, Cham, 2023b. ISBN 9783031371967. doi: 10.1007/978-3-031-37196-7_2. URL https://doi.org/10.1007/978-3-031-37196-7_2.

Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, June 1998. ISSN 1476-4687. doi: 10.1038/30918. URL https://www.nature.com/articles/30918.

Sajith Wettewa, Lei Hou, and Guomin Zhang. Graph Neural Networks for building and civil infrastructure operation and maintenance enhancement. *Advanced Engineering Informatics*, 62:102868, October 2024. ISSN 14740346. doi: 10.1016/j.aei.2024.102868. URL https://linkinghub.elsevier.com/retrieve/pii/S1474034624005160.

Richard C. Wilson and Ping Zhu. A study of graph spectra for comparing graphs and trees. *Pattern Recognition*, 41(9):2833–2841, 2008. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2008.03.011. URL https://www.sciencedirect.com/science/article/pii/S0031320308000927.

Sam F. L. Windels, Noël Malod-Dognin, and Nataša Pržulj. Combining graphlets and random walks for capturing complex network topology. *Scientific Reports*, March 2026. ISSN 2045-2322. doi: 10.1038/s41598-026-44410-x. URL https://www.nature.com/articles/s41598-026-44410-x.

Chen Xinhui, Yiwen Yuan, Joseph Liu, Chak Tou Leong, Xiaoye Zhu, and Jiaqi Chen. Generative models in protein engineering: A comprehensive survey. In *Neurips 2024 Workshop Foundation Models for Science: Progress, Opportunities, and Challenges*, 2024. URL https://openreview.net/forum?id=Xc7l84S0Ao.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Ömer Nebil Yaveroğlu, Noël Malod-Dognin, Darren Davis, Zoran Levnajic, Vuk Janjic, Rasa Karapandza, Aleksandar Stojmirovic, and Nataša Pržulj. Revealing the Hidden Language of Complex Networks. *Scientific Reports*, 4(1):4547, April 2014. ISSN 2045-2322. doi: 10.1038/srep04547. URL https://www.nature.com/articles/srep04547.

Jiaxuan You, Rex Ying, Xiang Ren, William Hamilton, and Jure Leskovec. GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 5708–5717. PMLR, July 2018. URL https://proceedings.mlr.press/v80/you18a.html.

Xitong Zhang, Yixuan He, Nathan Brugnone, Michael Perlmutter, and Matthew Hirn. MagNet: A Neural Network for Directed Graphs. In *Advances in Neural Information Processing Systems*, volume 34, pp. 27003–27015. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/hash/e32084632d369461572832e6582aac36-Abstract.html.

A NOTATION

Symbol	Description
$G = (V, E)$	Graph with node set V, edge set E, $n = \|V\|$
$o_i \in \mathbb{R}^n$	Orbit count vector: entry u is the count of orbit o_i at node u
$r(\cdot)$	Elementwise (mid-)rank transform
$\tilde{o}_i$	Mean-centered rank vector of orbit o_i
$\rho_s(o_i, o_j)$	Spearman (graphlet) correlation between orbits o_i and o_j
$A_{o_a\text{-}o_b}$	Orbit adjacency matrix: entry (u, v) counts co-occurrences of u at o_a, v at o_b
$H_{o_a\text{-}o_b}$	Haar-orbit adjacency: Hermitian matrix derived from $A_{o_a\text{-}o_b}$
$D_{o_a\text{-}o_b}$	Diagonal degree matrix: $D_{o_a\text{-}o_b}(u,u) = \sum_w \|H_{o_a\text{-}o_b}(u,w)\|$
$\tilde{H}_{o_a\text{-}o_b}$	Normalized Haar-orbit adjacency: $D^{-1/2}_{o_a\text{-}o_b} H_{o_a\text{-}o_b} D^{-1/2}_{o_a\text{-}o_b}$
$\hat{H}_{o_a\text{-}o_b}$	Spectrally filtered, positive semi-definite inner product matrix
$\rho_{loc}(o_i, o_j \mid \hat{H}_{o_a\text{-}o_b})$	Localized graphlet correlation of o_i, o_j conditioned on $\hat{H}_{o_a\text{-}o_b}$
k	Number of dominant eigenvectors retained in spectral filter

Table 2: Summary of mathematical notation used in the methods section.

B DATA

All reference datasets and generated graph sets are provided by the PolyGraph benchmark (Krimmel et al., 2026), which also provides outputs from four generative models: AutoGraph, DiGress, ESGG, and GRAN. The three datasets stress-test generative models at different structural scales: local topology (LOBSTER-L), global constraints (PLANAR-L), and mesoscale organisation (SBM-L). Each dataset contains 4,096 training and 4,096 test graphs. Network statistics are summarised in Table 3.

Lobster-L consists of tree-shaped lobster graphs, introduced as a graph generation benchmark by (You et al., 2018). Their sparse, regular branching structure makes them sensitive to failures in local topology: a model that correctly reproduces marginal degree distributions may still fail to reproduce the correct backbone-to-limb attachment patterns. Graphs are generated using `nx.random_lobster` with an expected number of backbone nodes of 80 and attachment probabilities 0.7 for both the backbone and its neighbours.

Planar-L consists of connected planar graphs, introduced as a benchmark by (Martinkus et al., 2022). Planarity is a global structural constraint that generative models tend to violate even when local statistics appear correct, making it a sensitive test of global topological fidelity. Graphs are generated by uniformly sampling 64 node positions in the unit square and computing the Delaunay triangulation via `scipy.spatial.Delaunay`.

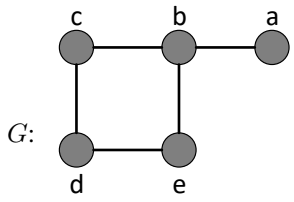
SBM-L consists of stochastic block model graphs, introduced as a benchmark by (Martinkus et al., 2022) to test whether models capture mesoscale community organisation. The sharp contrast between intra-community (0.3) and inter-community (0.005) edge probabilities means a model must simultaneously capture community membership and the correct edge density regime within and across communities. The number of communities is sampled uniformly from 2 to 5, with nodes per community from 20 to 40.

Table 3: Network statistics of the three benchmark datasets. Each entry reports the minimum, mean, and maximum over all graphs in the reference set. $|V|$: number of nodes; $|E|$: number of edges; Density: edge density $2|E|/(|V|(|V|-1))$; Diameter: longest shortest path; p_{90} SP: 90th percentile of pairwise shortest path lengths.

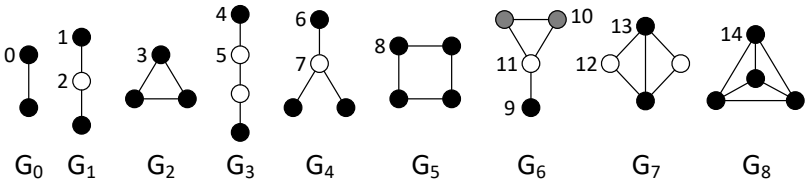
Dataset	$\|V\|$			$\|E\|$			Density			Diameter			p_{90} SP		
	min	mean	max	min	mean	max	min	mean	max	min	mean	max	min	mean	max
Planar	64	64.0	64	171	178.2	184	0.0848	0.0884	0.0913	6	7.1	9	5.0	5.3	6.0
SBM	40	105.1	187	106	499.9	1112	0.0558	0.1001	0.1880	3	5.6	10	2.0	4.2	8.0
Lobster	10	55.3	100	9	54.3	99	0.0200	0.0515	0.2000	2	9.7	27	2.0	7.9	24.0

C GRAPHLETS CONCEPTUAL ILLUSTRATIONS

A)



B)



C)

Graphlet	G_0	G_1		G_2	G_3		G_4		G_5	G_6			G_7		G_8
Orbit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
GDV(a)	1	2	0	0	2	0	1	0	0	0	0	0	0	0	0

Graphlet Count(a)	1	2	0	2	1	0	0	0	0

D)

$$A = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$A_{o_0\text{-}o_0}$

$$A_{o_1\text{-}o_2} = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{bmatrix} 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$$A_{o_1\text{--}o_1} = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 0 & 0 & 2 \\ 0 & 2 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 \end{bmatrix} \end{array}$$

Figure 4: **An illustration of graphlets, orbits, graphlet adjacency and orbit adjacency (modified from (Windels et al., 2026)). A)** Example network G. **B)** All graphlets with up to 4-nodes, labelled from G_0 to G_8. The automorphism orbits are indicated by the same shade and labelled from 0 to 14. **C):** The frequency at which node a occurs on each orbit in the example network G (panel A) and how those counts sum into graphlet counts. For instance, node a occurs on orbit o_1 twice: once in the path a-b-c and once in the path a-b-e. It never occurs on orbit o_2, i.e., the center of a three-node path. Hence, node a occurs on graphlet G_1 twice. **D)** The orbit adjacency matrices $A_{o_0\text{-}o_0}$, $A_{o_1\text{-}o_2}$ and $A_{o_1\text{--}o_1}$ for the example network G. The off-diagonal elements of $A_{o_1\text{-}o_2}$ correspond to the frequency at which the nodes in the corresponding rows and columns co-occur on graphlet G_1 in G, with the row-node touching orbit 1 and the $column$-node touching orbit 2. For instance, $A_{o_1\text{-}o_2}(a, b) = 2$, since a and b co-occur twice on G_1, with node a on orbit 1 and node b on orbit 2: via paths a-b-c and a-b-e. Analogously, $A_{o_1\text{--}o_1}(a, b) = 0$, since a and b never co-occur on G_1, with both a and b on orbit 1.

D ORBIT ADJACENCIES FOR FOUR-NODE GRAPHLETS

For a graph G, the orbit adjacency matrix $A_{o_a\text{-}o_b}$ is an $n \times n$ matrix whose entry $A_{o_a\text{-}o_b}(u, v)$ counts the number of times nodes u and v co-occur in orbits o_a and o_b respectively within the same induced graphlet instance (with $A_{o_a\text{-}o_b}(u, u) = 0$). The notation $A_{o_a\text{-}o_b}$ and $A_{o_a\text{--}o_b}$ distinguishes whether the two orbits are adjacent or at distance two within the graphlet. Figure 5 illustrates all twenty-eight such

orbit adjacencies for up to four-node graphlets (Windels et al., 2026), which serve as the conditioning structure for localized graphlet correlation (Section 2.1).

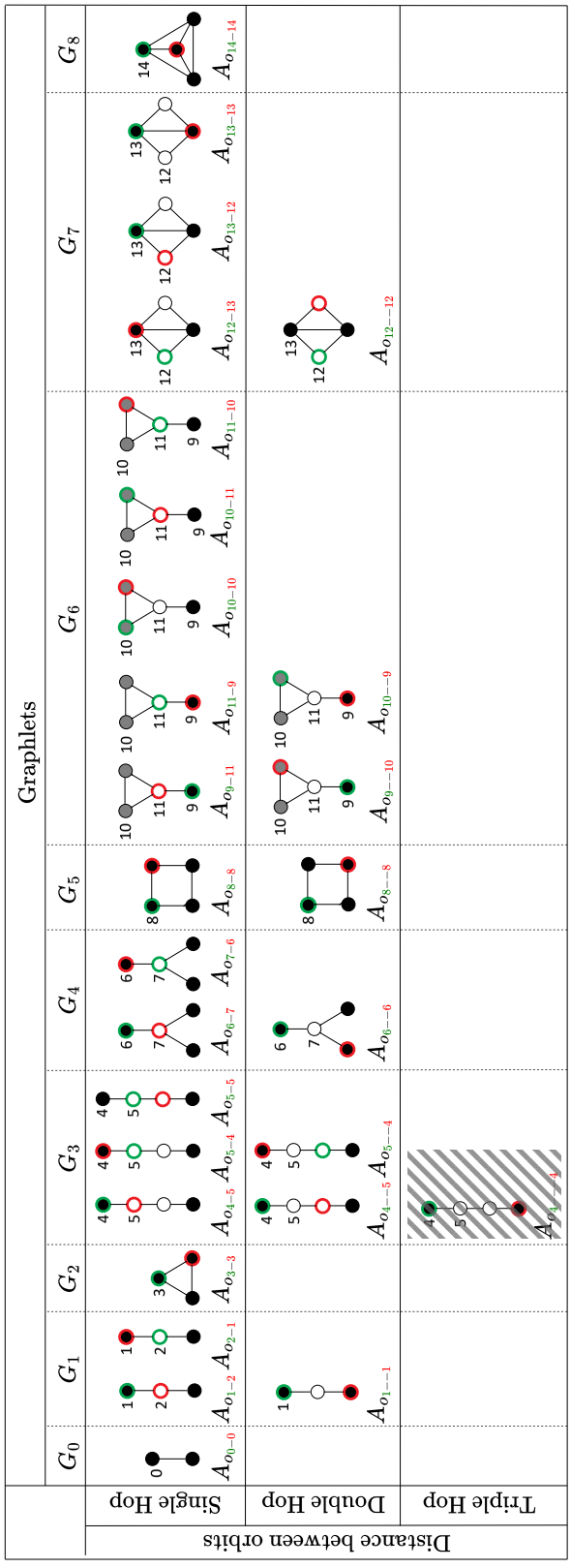


Figure 5: The twenty-eight orbit adjacencies for up to four-node graphlets. Each graphlet is shown with a pair of orbits (o_a, o_b) highlighted, indicating which node positions can co-occur in those orbits within the same graphlet instance.

E ABLATION STUDY: RECOMMENDED CONFIGURATION

We ablate each design choice against the full concatenated model (20 conditioned GCMs, Haar symmetrization, top-k filter, degree normalization, rank transform — 1540 features). Each ablation varies one component while keeping all others fixed. Table 4 reports PGD scores across all 12 model–dataset combinations with worst-case regret.

- **Degree normalization.** Removing degree normalization collapses SBM-L signal entirely (AutoGraph drops from 24.2 to 1.2; max regret 35.0).
- **Rank transform.** Rank-transformed orbit counts are more robust than raw counts (max regret 12.0 vs 17.3). Raw counts score higher on SBM-L (36.2 vs 24.2) but substantially lower on LOBSTER-L (79.1 vs 90.5).
- **Haar symmetrization.** Haar symmetrization outperforms simple symmetrization in 10 of 12 settings (up to $+5.0$ on PLANAR-L), with small losses on 2 SBM-L settings (up to -2.7).
- **Spectral filter.** The top-k filter (Section 2.3) is the most robust choice; $|\lambda|$ and λ^2 filters achieve higher scores on most datasets but collapse on SBM-L (max regret 34.7 and 35.3, respectively).
- **Generating set (6 adjacencies).** Reducing from 20 to 6 conditionings yields comparable robustness (max regret 12.4 vs 12.0), while outperforming on SBM-L (AutoGraph: 27.5 vs 24.2; DiGress: 36.8 vs 33.3).

Overall, we recommend the generating set of 6 adjacencies as the default configuration (median regret 3.2, max regret 12.4), with top-k filtering and degree normalization (both essential for SBM-L signal), and rank transform (more robust: max regret 12.0 vs 15.3 without). Since the 5 generating orbit adjacencies are all symmetric, the Haar construction simplifies to the standard normalized adjacency, requiring only 5 eigendecompositions (vs 19 for the full model).

Table 4: Ablation study. Each row varies one component from the full concatenated model (all 20 GCMs, Haar symmetrization, top-k filter, degree normalization, rank transform). PGD scores ($\times 100$, mean $\pm$ std over 10 subsamples) are reported per model–dataset combination. The most discriminative setting per combination is shown in **bold**, the second-most is underlined. Median and max regret summarise robustness across all 12 combinations.

Setting	PLANAR-L				LOBSTER-L				SBM-L				Regret	
	AutoGraph	DIGRESS	GRAN	ESGG	AutoGraph	DIGRESS	GRAN	ESGG	AutoGraph	DIGRESS	GRAN	ESGG	Median	Max
All 20 adjacencies	57.6 $\pm$ 1.1	43.0 $\pm$ 1.9	99.2 $\pm$ 1.1	60.7 $\pm$ 1.0	90.5 $\pm$ 0.8	95.3 $\pm$ 0.7	89.7 $\pm$ 0.7	90.3 $\pm$ 0.7	24.2 $\pm$ 1.4	33.3 $\pm$ 1.0	64.8 $\pm$ 1.0	94.7 $\pm$ 0.5	3.4	12.0
Raw counts (no rank)	49.3 $\pm$ 0.6	**47.3** $\pm$ 1.9	99.3 $\pm$ 0.7	53.2 $\pm$ 1.2	79.1 $\pm$ 0.8	79.9 $\pm$ 0.9	86.5 $\pm$ 0.6	86.9 $\pm$ 0.6	**36.2** $\pm$ 1.3	**43.1** $\pm$ 0.7	**68.2** $\pm$ 0.9	96.3 $\pm$ 0.4	2.5	17.3
Simple sym. (no Haar)	56.0 $\pm$ 0.9	38.0 $\pm$ 2.2	99.0 $\pm$ 1.3	56.6 $\pm$ 1.4	88.9 $\pm$ 0.7	94.4 $\pm$ 0.7	89.1 $\pm$ 0.7	88.9 $\pm$ 0.6	26.1 $\pm$ 1.6	36.1 $\pm$ 1.6	64.0 $\pm$ 1.1	94.6 $\pm$ 0.4	4.6	10.1
$\lvert\lambda\rvert$ filter	58.2 $\pm$ 1.1	42.2 $\pm$ 2.0	99.5 $\pm$ 0.6	61.2 $\pm$ 1.3	92.6 $\pm$ 0.6	95.7 $\pm$ 0.6	90.2 $\pm$ 0.6	91.2 $\pm$ 0.5	1.5 $\pm$ 1.3	11.9 $\pm$ 2.2	64.5 $\pm$ 1.0	94.5 $\pm$ 0.4	2.8	34.7
λ^2 filter	**61.9** $\pm$ 1.1	42.2 $\pm$ 2.0	**99.5** $\pm$ 0.6	**64.2** $\pm$ 1.2	**93.9** $\pm$ 0.5	**97.2** $\pm$ 0.5	**90.7** $\pm$ 0.6	**92.5** $\pm$ 0.4	0.9 $\pm$ 1.5	13.4 $\pm$ 2.0	65.3 $\pm$ 1.2	95.9 $\pm$ 0.5	0.0	35.3
No degree norm.	52.6 $\pm$ 1.0	41.0 $\pm$ 1.7	99.2 $\pm$ 1.2	56.5 $\pm$ 1.3	87.3 $\pm$ 0.6	93.1 $\pm$ 0.7	89.7 $\pm$ 0.8	90.7 $\pm$ 0.6	1.2 $\pm$ 2.0	11.2 $\pm$ 1.6	56.4 $\pm$ 1.0	**97.0** $\pm$ 0.5	6.4	35.0
1 adjacency	43.2 $\pm$ 1.2	32.1 $\pm$ 2.4	99.2 $\pm$ 1.0	45.8 $\pm$ 1.4	82.5 $\pm$ 1.4	89.6 $\pm$ 1.0	87.9 $\pm$ 0.8	85.4 $\pm$ 0.9	2.9 $\pm$ 2.6	9.4 $\pm$ 2.1	61.5 $\pm$ 0.7	93.0 $\pm$ 0.3	9.5	33.7
4 adjacencies	53.9 $\pm$ 1.4	34.7 $\pm$ 2.3	99.0 $\pm$ 1.4	57.0 $\pm$ 1.3	89.6 $\pm$ 1.1	94.8 $\pm$ 0.6	89.3 $\pm$ 0.9	89.1 $\pm$ 0.6	6.1 $\pm$ 3.0	14.4 $\pm$ 1.5	63.4 $\pm$ 1.1	94.2 $\pm$ 0.5	4.6	30.1
13 adjacencies	60.7 $\pm$ 0.9	41.9 $\pm$ 2.3	99.2 $\pm$ 0.8	61.5 $\pm$ 1.1	89.2 $\pm$ 1.0	94.7 $\pm$ 0.8	89.5 $\pm$ 0.8	89.3 $\pm$ 0.6	25.3 $\pm$ 1.2	34.7 $\pm$ 1.5	64.9 $\pm$ 0.8	94.3 $\pm$ 0.4	3.0	10.9
5 generating adjacencies	54.0 $\pm$ 1.7	34.9 $\pm$ 2.3	99.0 $\pm$ 1.4	53.0 $\pm$ 1.6	87.8 $\pm$ 1.1	92.8 $\pm$ 1.0	88.3 $\pm$ 0.8	88.4 $\pm$ 0.6	27.5 $\pm$ 1.9	36.8 $\pm$ 1.4	63.8 $\pm$ 1.0	94.3 $\pm$ 0.4	5.3	12.4

F FEATURE IMPORTANCE ANALYSIS

To identify which orbit adjacencies drive the discriminative performance reported in Section 3.1, we perform two-stage grouped permutation importance on the concatenated Assort. descriptor. In Stage 1 (Table 5), all features from one orbit adjacency are shuffled simultaneously, quantifying the contribution of each conditioning to the overall PGD score. In Stage 2 (Table 6), individual features within the top orbit adjacency are permuted to identify the most informative orbit pairs.

On SBM-L, where Assort. uniquely outperforms all other descriptors, the $A_{o_{14}\text{-}o_{14}}$ conditioned GCM accounts for 58% (AutoGraph) and 62% (DiGress) of the discriminative signal, detecting disrupted clique co-occurrence structure invisible to the unconditioned GCM (identity, 2nd at 18% and 12% respectively). Within this adjacency, the signal is distributed across multiple orbit pairs rather than concentrated in a single one (Table 6). On LOBSTER-L, $A_{o_0\text{-}o_0}$ (edge co-occurrence) dominates for AutoGraph (75%) and DiGress (59%), capturing local branching structure. Conversely, when the identity already captures the discriminative signal, as for GRAN and ESGG on LOBSTER-L (89% and 98%), conditioning adds little, explaining why Assort. offers the largest advantage precisely where existing descriptors fail.

Table 5: Stage 1: grouped permutation importance by orbit adjacency. For each model–dataset combination, we report the two orbit adjacencies contributing most to the PGD score, with their percentage contribution. GRAN on PLANAR-L is trivially separable (PGD ≈ 100).

Dataset	Model	Top adjacency	%	2nd adjacency	%
PLANAR-L	AutoGraph	$A_{o_{14}\text{-}o_{14}}$	30.9	$A_{o_0\text{-}o_0}$	24.5
	DIGRESS	$A_{o_0\text{-}o_0}$	27.7	$A_{o_{12}\text{-}\text{-}o_{12}}$	20.2
	GRAN	(trivially separable)			
	ESGG	I	36.4	$A_{o_{14}\text{-}o_{14}}$	20.6
LOBSTER-L	AutoGraph	$A_{o_0\text{-}o_0}$	75.4	I	16.0
	DIGRESS	$A_{o_0\text{-}o_0}$	59.4	I	38.2
	GRAN	I	89.0	$A_{o_0\text{-}o_0}$	9.0
	ESGG	I	98.3	$A_{o_4\text{-}\text{-}o_5}$	0.8
SBM-L	AutoGraph	$A_{o_{14}\text{-}o_{14}}$	58.3	I	17.5
	DIGRESS	$A_{o_{14}\text{-}o_{14}}$	62.3	I	11.9
	GRAN	I	60.8	$A_{o_4\text{-}o_5}$	10.1
	ESGG	I	86.8	$A_{o_6\text{-}o_7}$	5.0

Table 6: Stage 2: individual feature importance within the top orbit adjacency. The three most informative orbit pairs are reported with their importance scores. "Redundant" indicates all features contribute approximately equally.

Dataset	Model	Adjacency	1st pair	2nd pair	3rd pair
PLANAR-L	AutoGraph	$A_{o_{14}\text{-}o_{14}}$	$(6,7)$=0.0324	$(5,7)$=0.0277	$(9,10)$=0.0099
	DIGRESS	$A_{o_0\text{-}o_0}$	$(5,10)$=0.1118	$(1,4)$=0.0379	$(6,10)$=0.0169
	GRAN	(trivially separable)			
	ESGG	I	$(9,10)$=0.0293	$(4,7)$=0.0230	$(5,9)$=0.0177
LOBSTER-L	AutoGraph	$A_{o_0\text{-}o_0}$		(redundant)	
	DIGRESS	$A_{o_0\text{-}o_0}$		(redundant)	
	GRAN	I	$(1,8)$=0.0007	$(8,10)$=0.0002	$(8,9)$=0.0001
	ESGG	I	$(5,9)$=0.0007	$(2,4)$=0.0000	$(1,8)$=0.0000
SBM-L	AutoGraph	$A_{o_{14}\text{-}o_{14}}$	$(6,7)$=0.1290	$(1,6)$=0.1126	$(1,9)$=0.1084
	DIGRESS	$A_{o_{14}\text{-}o_{14}}$	$(9,11)$=0.0377	$(1,2)$=0.0316	$(0,10)$=0.0208
	GRAN	I	$(2,4)$=0.0024	$(0,4)$=0.0012	$(9,10)$=0.0007
	ESGG	I		(redundant)	

G EDGE CASES AND NUMERICAL CONSIDERATIONS

Self-loops ($A + I$) ensure all node degrees are ≥ 1, preventing division by zero in the degree normalization of Equation 6. Constant orbit-count vectors produce zero denominators in Equation 6, which we handle identically to undefined standard correlation by setting the value to 0. Eigenvalue ties at the top-k boundary are resolved by the LAPACK ordering convention; results are stable across all benchmark settings.

H GRAPH-STRUCTURE DESCRIPTORS

The following structural descriptors are commonly used to evaluate graph generative models (Martinkus et al., 2022; Vignac et al., 2022; You et al., 2018; Chen et al., 2025), with average orbit counts being the best-performing empirically (Krimmel et al., 2026). We also include the GCM (Yaveroğlu et al., 2014), which extends average orbit counts by capturing correlations between orbit types, but has not previously been used in this context:

- **Degree distribution.** The degree distribution characterizes the distribution of node degrees, capturing degree heterogeneity and structural imbalances such as the presence of hubs or isolated nodes.
- **Clustering coefficient distribution.** The distribution of clustering coefficients measures, for each node, the ratio of connected neighbour pairs (i.e., forming a triangle with the node) to all possible neighbour pairs, capturing the tendency of the graph to form locally dense, clique-like structures (Watts & Strogatz, 1998).
- **Graph spectrum.** The graph spectrum is the sorted vector of eigenvalues of the normalized Laplacian (defined in Section 2.2), encoding global structural properties of the graph such as connectivity and regularity (Wilson & Zhu, 2008).
- **Average orbit counts.** For each node, orbit counts measure the frequency with which the node participates in each automorphism orbit of small induced subgraphs (graphlets) of size four or five nodes. These counts are subsequently averaged across all nodes, yielding a graph-level summary of local structural motifs and connectivity patterns. Considering up to four-node graphlets yields 15 orbits (**Orbit4**); considering up to five-node graphlets yields 73 orbits (**Orbit5**).
- **Graphlet Correlation Matrix (GCM).** The GCM describes a graph through the Spearman rank correlations between orbit count vectors of all pairs of non-redundant orbits — orbits whose counts cannot be derived from those of other orbits (Yaveroğlu et al., 2014). This yields $\binom{11}{2} = 55$ unique correlations for up to four-node graphlets (**GCM4**) and $\binom{56}{2} = 1540$ for up to five-node graphlets (**GCM5**).
- **Randomized GIN.** Thompson et al. (Thompson et al., 2022) showed that the GIN architecture (Xu et al., 2019), whose expressive power matches the Weisfeiler-Leman graph isomorphism test, serves as an effective structural descriptor for generative evaluation even without training. Using randomly initialised, untrained weights makes the descriptor fully domain-agnostic, avoiding task-specific biases from pretraining. The graph-level readout embedding serves as the descriptor (**GIN**).

I PROOF: LOCALIZED GRAPHLET CORRELATION IS BOUNDED IN $[-1, 1]$

Proposition I.1. *For any orbit count vectors o_i, o_j and orbit adjacency $\hat{H}_{o_a\text{-}o_b}$, the localized graphlet correlation satisfies* $\mathrm{Re}(\rho_{loc}), \mathrm{Im}(\rho_{loc}) \in [-1, 1]$.

Proof. Since $\hat{H}_{o_a\text{-}o_b}$ is positive definite, $\langle x, y\rangle = x^\top \hat{H}_{o_a\text{-}o_b}\, y$ defines a valid inner product. For non-zero $\tilde{o}_i$, $\tilde{o}_j$, positive definiteness gives $\tilde{o}_i^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_i > 0$ and $\tilde{o}_j^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j > 0$, so the denominator of ρ_{loc} is strictly positive. By the Cauchy-Schwarz inequality, $|\langle x, y\rangle|^2 \leq \langle x, x\rangle\langle y, y\rangle$, i.e.,

$$|\tilde{o}_i^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j|^2 \leq (\tilde{o}_i^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_i)(\tilde{o}_j^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j).$$

Table 7: PGD scores ($\times 100$, mean $\pm$ standard deviation over 10 subsamples) for all descriptors considered in this paper, obtained by comparing subsets of the same reference dataset drawn from its training and test splits. Since both subsets are drawn from the same distribution, the true JS distance is zero; a well-calibrated descriptor should yield a score ≈ 0.

Metric	PLANAR-L	LOBSTER-L	SBM-L
PGD	0.2 ± 0.5	0.0 ± 0.0	0.3 ± 0.7
Assort.	1.0 ± 1.6	0.2 ± 0.5	0.8 ± 1.9
GCM4	0.3 ± 0.5	0.2 ± 0.5	0.2 ± 0.7
GCM5	0.2 ± 0.5	0.0 ± 0.0	0.5 ± 1.0
Orb4.	0.2 ± 0.6	0.2 ± 0.4	1.1 ± 1.0
Orb5.	0.3 ± 0.5	0.2 ± 0.4	0.3 ± 0.6
Deg.	0.4 ± 0.5	0.5 ± 0.8	1.1 ± 1.1
Spectral	0.4 ± 0.8	1.1 ± 1.5	0.3 ± 0.8
Clust.	0.5 ± 0.9	0.0 ± 0.0	0.9 ± 1.3
GIN	0.1 ± 0.3	0.3 ± 0.5	0.8 ± 0.8

Dividing both sides by $(\tilde{o}_i^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_i)(\tilde{o}_j^\top \hat{H}_{o_a\text{-}o_b}\, \tilde{o}_j) > 0$ gives $|\rho_{loc}|^2 \leq 1$. Since $|\rho_{loc}| \geq 0$, taking square roots gives $|\rho_{loc}| \leq 1$. The claim then follows from $|\mathrm{Re}(z)| \leq |z|$ and $|\mathrm{Im}(z)| \leq |z|$ for any $z \in \mathbb{C}$. □

J SANITY CHECK: PGD CALIBRATION ON REFERENCE GRAPHS

To verify that PGD is well calibrated for each descriptor, (Krimmel et al., 2026) compute PGD on two non-overlapping subsets of the same reference dataset, one drawn from its training split and one from its test split. For each of the three datasets (PLANAR-L, LOBSTER-L, and SBM-L; 8,192 training and 4,096 test graphs each), 2,048 graphs are drawn without replacement from each split, and this is repeated 10 times. Since both subsets are drawn from the same reference dataset, the true JS distance between them is zero and PGD should therefore be ≈ 0 regardless of the descriptor used. We replicate this experiment for all descriptors considered in this paper in Table 7.

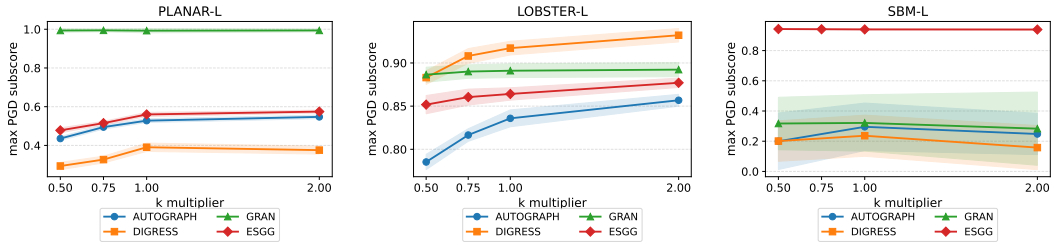
For the existing descriptors, scores are consistent with those reported in (Krimmel et al., 2026). Our new descriptors (Assort., GCM4, and GCM5) also yield scores close to zero on the three procedural datasets, ranging from 0.0 to 1.0, and are generally comparable to or lower than the existing descriptors. The exception is Assort. on PLANAR-L, which scores 1.0 compared to a maximum of 0.5 among existing descriptors on that dataset. Together, these results confirm that Assort., GCM4, and GCM5 are well calibrated and can be used as descriptors within the PGD framework.

K ROBUSTNESS TO THE NUMBER OF DOMINANT EIGENVECTORS k

Section 3.1 benchmarks graph generative models using Assort. as a PGD subscore, with $k = \sqrt{n/2}$ following (von Luxburg, 2007). Here we verify that the results reported there are robust to this choice by multiplying k by 0.5, 0.75, 1, and 2.

For each combination of dataset, model, and k multiplier, we run PolyGraph over all 20 conditioned GCMs and report the maximum PGD subscore, with error bars showing its standard deviation over 10 bootstrap subsamples. Figure 6 shows the results.

Overall, $k = \sqrt{n/2}$ appears to be a reasonable default. On PLANAR-L and LOBSTER-L, scores increase monotonically from $\times 0.5$ to $\times 1$, and continue to rise slightly beyond that on LOBSTER-L, suggesting that retaining more eigenvectors can still yield marginal gains on these datasets. On SBM-L, scores at $k > \sqrt{n/2}$ are stable or decrease slightly, so the default is already at or beyond the useful range. On SBM-L, AutoGraph and DiGress show large error bars across all k values — consistent with the high variance already observed in Table 1. Comparing the Assort. PGD scores shown here with the scores from Table 1, Assort. remains the best-performing in 6 out of 12 model–dataset combinations across all values of k.

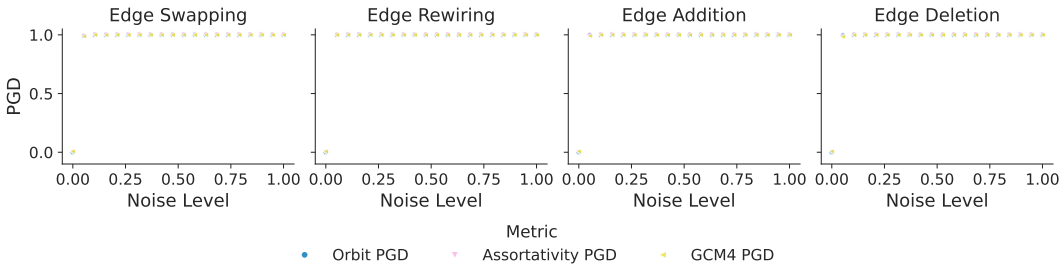


(a) PLANAR-L (b) LOBSTER-L (c) SBM-L

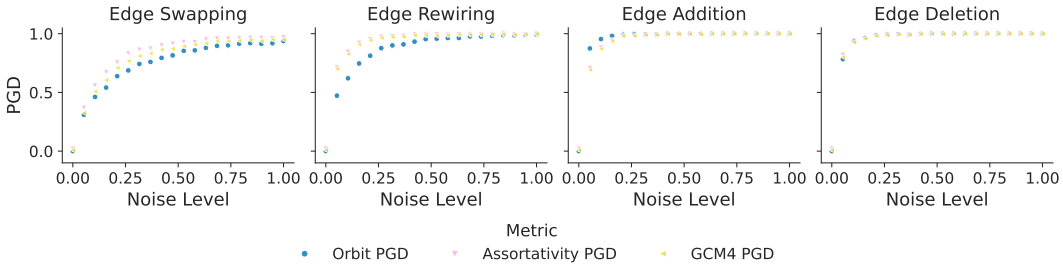
Figure 6: PGD subscore for Assort. as a function of the number of dominant eigenvectors k retained in the spectral filter, across all three datasets.

L DESCRIPTOR SENSITIVITY TO STRUCTURAL PERTURBATIONS

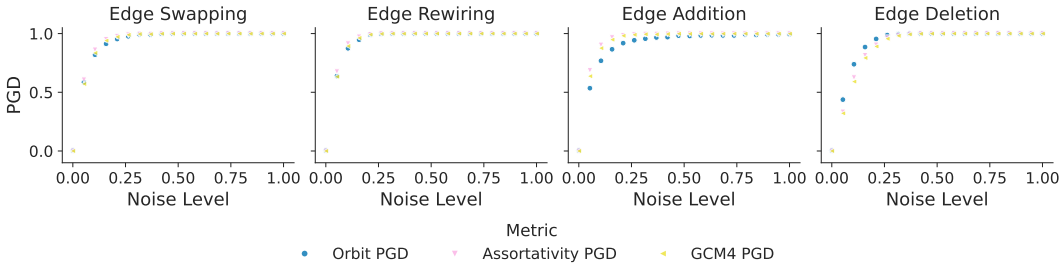
Section 3.2 uses controlled perturbation experiments on PLANAR-L to explain why correlation-based descriptors outperform orbit counts: they detect perturbations that leave marginal orbit count distributions unchanged. Here we show the full results across all three datasets (PLANAR-L, LOBSTER-L, SBM-L) and all descriptors, confirming that the pattern generalises.



(a) PLANAR-L



(b) LOBSTER-L



(c) SBM-L

Figure 7: PGD as a function of noise level p for all descriptors across three datasets and four perturbation types.

M COMPUTATIONAL COST

Section 3.3 analyses the trade-off between discriminatory power and computational cost for AutoGraph. Here we provide the full results across all four generative models. For each descriptor and model, we plot PGD subscore against median per-graph computation time (in milliseconds, measured over the reference test set on a single CPU node: Intel Xeon Silver 4116 @ 2.10 GHz, 8 cores, 8 GB RAM).

The pattern observed for AutoGraph holds broadly across models: Assort. is consistently the most expensive descriptor, yet 1.5–2.9$\times$ slower than GIN, the next slowest. On SBM-L, Assort. is the only descriptor that recovers meaningful signal for AutoGraph and DiGress, while on LOBSTER-L it is the best-performing descriptor across all models. GCM4 is consistently fast (near Orbit4 speed) while outperforming orbit counts on LOBSTER-L.

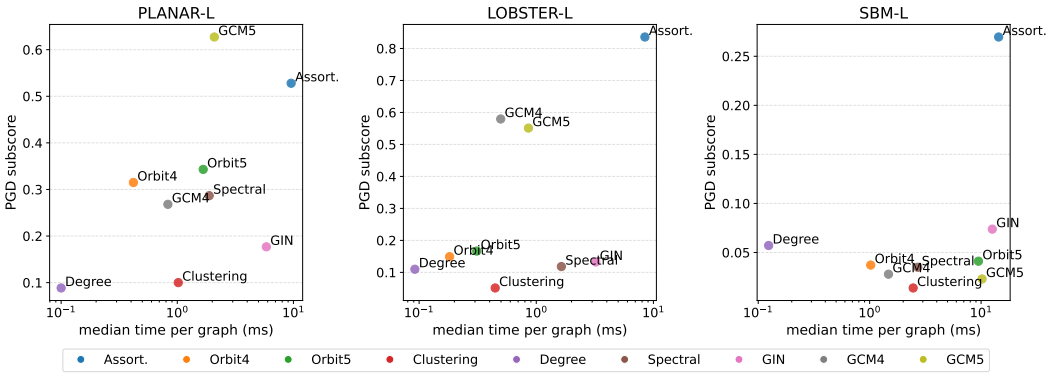


Figure 8: Per-descriptor discriminatory power (PGD subscore) vs. median per-graph computation time (ms, over reference graphs) for AutoGraph. Each subplot corresponds to one dataset (top row: PLANAR-L, LOBSTER-L; bottom row: SBM-L).

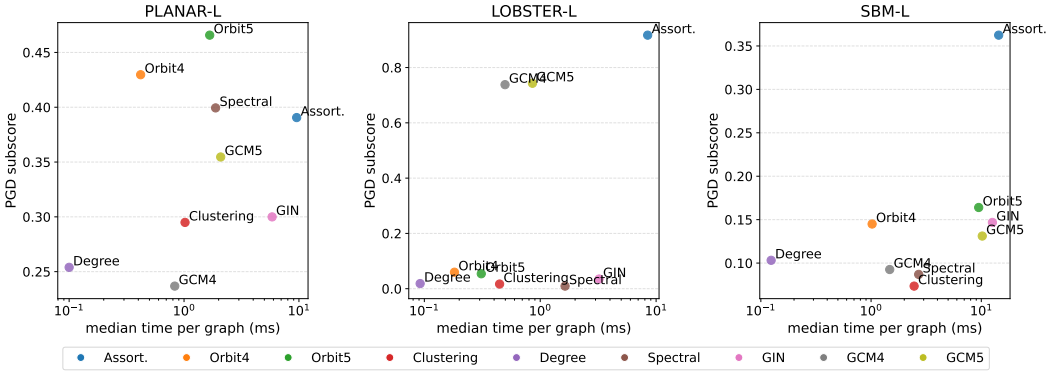


Figure 9: Per-descriptor discriminatory power (PGD subscore) vs. median per-graph computation time (ms, over reference graphs) for DiGress. Each subplot corresponds to one dataset (top row: PLANAR-L, LOBSTER-L; bottom row: SBM-L).

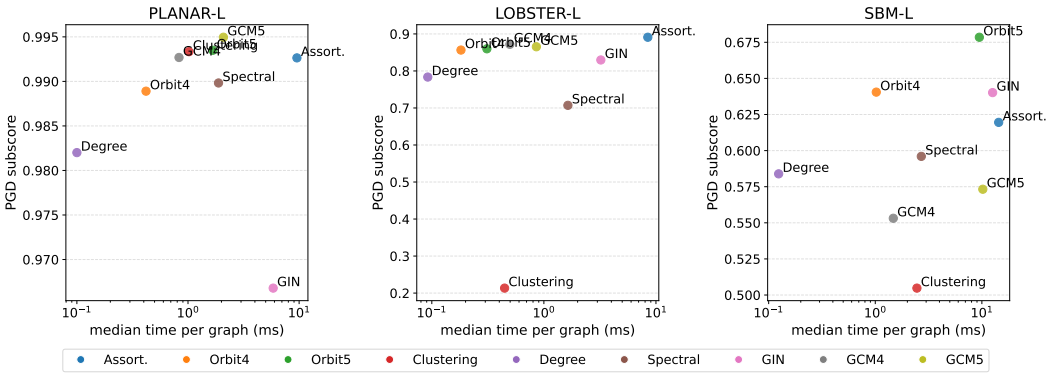


Figure 10: Per-descriptor discriminatory power (PGD subscore) vs. median per-graph computation time (ms, over reference graphs) for GRAN. Each subplot corresponds to one dataset (top row: PLANAR-L, LOBSTER-L; bottom row: SBM-L).

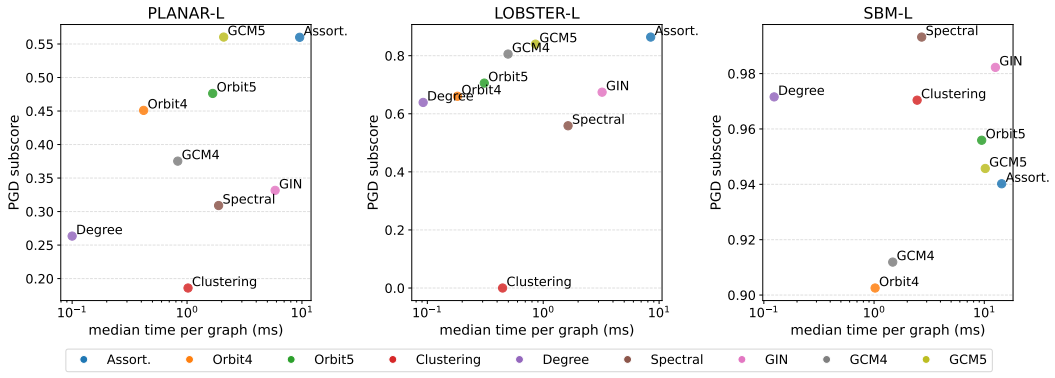


Figure 11: Per-descriptor discriminatory power (PGD subscore) vs. median per-graph computation time (ms, over reference graphs) for ESGG. Each subplot corresponds to one dataset (top row: PLANAR-L, LOBSTER-L; bottom row: SBM-L).