A GENERATIVE MODEL OF COMPLEX NETWORKS USING GRAPHONS AND NEURAL INVERSE OPERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative graph models are central to understanding and simulating complex networks. However, existing approaches have complementary strengths and limitations. Mechanistic models offer interpretability but rely on instance-specific estimation methods. Deep generative models, on the other hand, offer amortized inference at the cost of interpretability and are largely limited to graph sizes seen during training. Scientific applications motivate a framework that retains the strengths of both paradigms. We bridge them by formulating both the generative model and parameter recovery in function space. A multifractal step graphon extends standard step graphons with a recursive construction that compactly parameterizes complex networks. This formulation admits a neural inverse operator to recover its parameters, enabling inference on unseen graph sizes. We evaluate our model, trained only on synthetic multifractal step graphon realizations, against both paradigms. Against a graph foundation model pretrained on empirical networks, our method achieves the best average performance on three of four metrics in a zero-shot graph-generation benchmark, indicating that the model transfers to real-world graphs. We also apply our method to single-observation networks, a regime largely inaccessible to deep models that require training corpora, where it performs comparably to an instance-specific method that optimizes on each graph. In a multi-subject EEG case study, the inferred parameters track a reversible change in brain state more sensitively than traditional network statistics. Together, these results indicate that mechanistic interpretability and amortized inference can be effectively unified in a generative graph model to enhance our understanding of complex networks.

1 INTRODUCTION

Complex networks model biological (La Rocca et al., 2018; Yin et al., 2020; Yang et al., 2021), social (Lee & Jung, 2006; Liu et al., 2017), and infrastructure systems (Long & Chen, 2021; Batty & Longley, 1994). Summary statistics of such networks aggregate features across scales and can obscure differences in the underlying structure (Xue & Bogdan, 2017). In addition, these metrics can vary spuriously across different partial observations of the same network (Lee et al., 2006; Stumpf et al., 2005; Achlioptas et al., 2005). This motivates going beyond descriptive statistics and recovering the compact generative mechanisms that reproduce the observed network. Such mechanisms are particularly valuable for revealing hidden differences underlying complex networks (Yang & Bogdan, 2020; Akarca et al., 2021). As such, recovering these generative processes is essential for enhancing our understanding of complex systems.

Generative graph models synthesize realistic networks by capturing meaningful representations of their underlying structure. Two paradigms have emerged, each with its own approach. Mechanistic models define explicit wiring rules that are algorithmically implemented to produce graphs with prescribed properties (Erdős & Rényi, 1960; Barabási & Albert, 1999). In contrast, deep generative models learn graph distributions directly from data (Li et al., 2018; Kipf & Welling, 2016).

[1]Figure 1 incorporates a brain tractography image by Yeh et al. (2018) [CC BY-SA 4.0] and a protein interaction network by Chen et al. (2019) [CC BY 4.0] via Wikimedia Commons. The citation network by citationgraph.org is Public Domain (CC0). This derived schematic is distributed under CC BY-SA 4.0.

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.

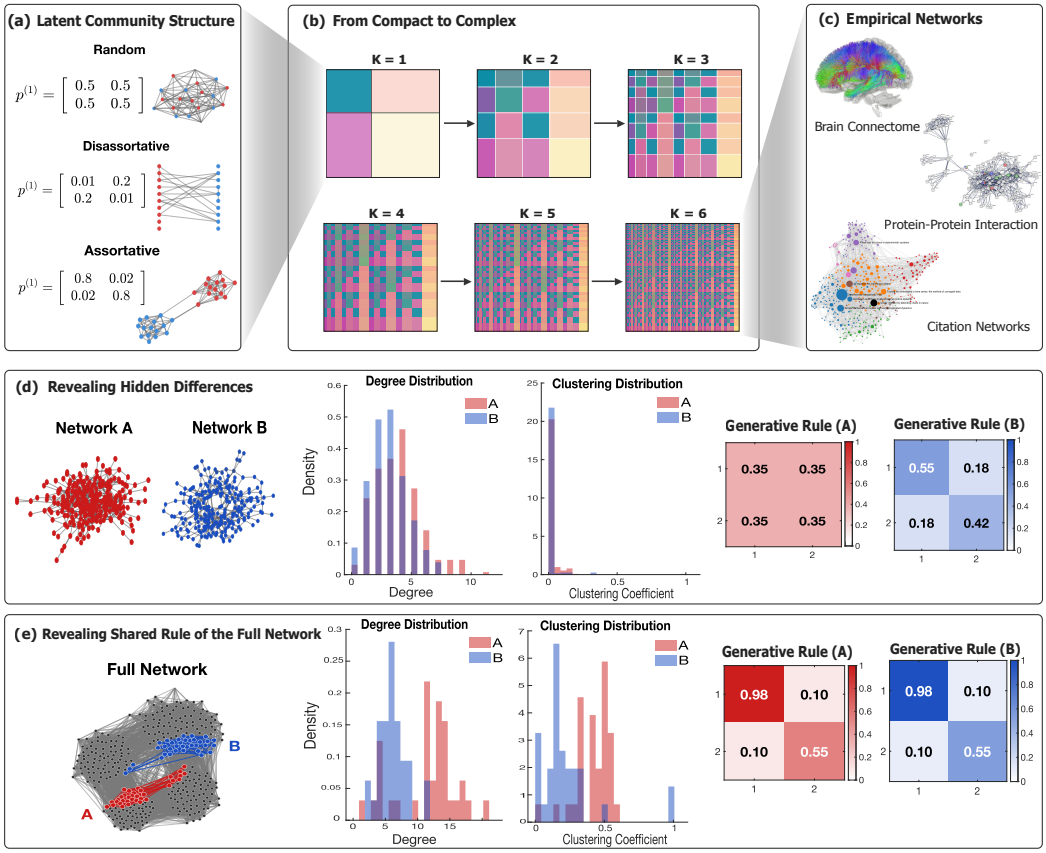

Figure 1: **Discovery of Hidden Structures in Real-World Networks (a)** The base graphon encodes hidden community structures underlying complex networks. **(b)** Kronecker expansion of base parameters produces the multifractal step graphon. **(c)** Finite realization of the graphon produces networks that share similar properties with those seen in various empirical domains [1]. Recovering the generative rule reveals **(d)** hidden differences between networks with the same statistics and **(e)** the shared latent structure of a network from different partial observations, where subsampling can produce different statistics.

Classical mechanistic models capture specific regimes, from homogeneous random graphs (Erdős & Rényi, 1960) to small-world (Watts & Strogatz, 1998) and preferential-attachment networks (Barabási & Albert, 1999). However, these frameworks are restricted to modeling a prescribed class of networks. To overcome these constraints, recursive models such as SKG (Leskovec et al., 2010), multifractal network generators (Palla et al., 2010), and WMGM (Yang et al., 2021) established the foundation for using recursive rules to compactly describe networks observed across many domains.

However, recovering these models from data remains challenging. SKGs (Leskovec et al., 2010) rely on maximum likelihood estimation (MLE) with permutation sampling. Multifractal network generators (Palla et al., 2010) use simulated annealing, while the weighted multifractal generative model (WMGM) (Yang et al., 2021) relies on expectation-maximization (EM). MLE and EM are susceptible to local optima (Wu, 1983; Blei et al., 2017), whereas simulated annealing suffers from prohibitively slow convergence (Hajek, 1988). These methods also search over latent node-to-community assignments, a combinatorial step whose cost grows rapidly with graph size (Leskovec et al., 2010; Daudin et al., 2008; Brandes et al., 2008). Moreover, these methods are instance-specific, so the aforementioned costs recur with every observation.

Deep generative graph models (Simonovsky & Komodakis, 2018; You et al., 2018; Liao et al., 2019; Vignac et al., 2022) amortize generation from data, but their learned representations are often less interpretable and typically do not extrapolate to graph sizes outside the training distribution. Autoregressive models (Li et al., 2018; You et al., 2018; Liao et al., 2019) cannot extrapolate beyond the training distribution due to architectural constraints (e.g., GraphRNN's BFS window and

GRAN's maximum node capacity). Variational autoencoders and diffusion-based models similarly generate graphs whose sizes are effectively limited to those observed during training (Simonovsky & Komodakis, 2018; Niu et al., 2020; Vignac et al., 2022).

This work. Taken together, there is a strong need for a framework that retains the strengths of both paradigms, such as interpretability and amortized inference, which are jointly necessary for generative models to serve in scientific applications. Generalization to unseen graph sizes further broadens its applicability. We address this need by formulating both the generative model and its inference in function space:

- **Compact Function-Space Representation of Complex Networks:** We introduce the multifractal step graphon: a recursively-refined, piecewise-constant kernel parameterized by a small set of base coefficients $\theta^{(1)} = (l^{(1)}, p^{(1)})$ that directly encode hierarchical community structure.
- **Neural Inverse Operator Formulation:** We cast the parametric graphon recovery as an inverse operator-learning problem. This formulation enables amortized inference that generalizes to unseen graph sizes.

2 Related Work

Graphons. A graphon is a bounded, measurable function $W : [0, 1]^2 \to [0, 1]$ encoding connection probabilities between latent node positions (Lovász & Szegedy, 2006; Borgs et al., 2008). Step graphons approximate this continuous space as piecewise-constant surfaces, admitting an interpretable parametrization in which latent communities and connection rules correspond to domain partitions and uniform probabilities within each block, respectively (Airoldi et al., 2013). Prior work on graphon estimation has been limited to instance-specific statistical methods (Airoldi et al., 2013; Chan & Airoldi, 2014). Moreover, standard step graphons implicitly assume non-hierarchical topologies and cannot represent the recursive, self-similar structure present in many real-life networks (Xue & Bogdan, 2017).

Neural Operators. Recently, neural operators have emerged as powerful frameworks for learning the mapping between function spaces, with applications primarily in solving forward partial differential equations (Lu et al., 2021; Kovachki et al., 2023; Li et al., 2021; Gupta et al., 2021; Xiao et al., 2023). *Inverse* operator learning, in which the network recovers underlying parameters from functions, has been applied to PDE inverse problems (Molinaro et al., 2023). However, no other work has applied it directly to graphon estimation to the best of our knowledge.

Graph Generation. The complementary limitations of mechanistic and deep generative graph models are discussed in Section 1. Existing graphon-based generative models offer either interpretability, through a linear factorization over learned graphon factors in a graphon autoencoder (GNAE) (Xu et al., 2021) or flexibility across graph sizes by representing graphons as implicit neural representations at arbitrary resolution (Xia et al., 2023; Azizpour et al., 2024). Our approach provides both. The method's parameters define an explicit wiring rule with a closed-form link to network properties (Appendix B), whereas GNAE's interpretability lies in its learned latent factors. Its function-space formulation further admits a neural inverse operator that generalizes across graph sizes.

3 Multifractal Step Graphon Formulation

3.1 Overview

Here, we extend step graphons (Airoldi et al., 2013) to better represent hierarchical structure observed in real-life networks (Xue & Bogdan, 2017). We formalize this through a recursively refined piecewise-constant kernel on the unit square, which gives a continuum model parameterized by $\theta^{(1)} = (l^{(1)}, p^{(1)})$. The parameters $l^{(1)}$ and $p^{(1)}$ yield heterogeneous node attribution and weighted connectivity, respectively. M defines the number of base communities. A Kronecker recursion expands these base parameters to arbitrary depth K, partitioning the latent space into M^K communities whose sizes and interaction strengths factorize multiplicatively across scales. We use $M = 2$

throughout, which keeps the M^K communities populated in tractable training graphs. Because two recursion steps together form a valid $M = 4$ base rule, this choice still covers the subset of larger-M models whose base rule is Kronecker-factorizable (Appendix A).

We model generation of graphs from the multifractal step graphon as realizations of a jointly exchangeable array. By the Aldous representation theorem (Aldous, 1981), any such array admits a generative representation as a measurable function $W : [0, 1]^2 \to [0, 1]$ from which finite graphs are sampled. This provides the foundation for graphon-based generative modeling. Thus, our use of the graphon follows the Aldous tradition—as a generative kernel from which finite networks are sampled—rather than the Lovász–Szegedy tradition of graph limits.

3.1.1 Continuous Latent Space

A graphon is a bounded, Lebesgue-measurable function $W : [0, 1]^2 \to [0, 1]$ belonging to the Hilbert space $L^2([0, 1]^2)$ (Lovász & Szegedy, 2006; Borgs et al., 2008). While traditionally defined as symmetric ($W(x, y) = W(y, x)$) for undirected graphs, we follow theoretical (Aldous, 1981) and empirical (Airoldi et al., 2013) precedents and generalize to asymmetric kernels ($W(x, y) \neq W(y, x)$) to capture directed topologies. In this formulation, $W(x, y)$ represents the probability of a directed edge from a node at latent coordinate x to a node at y.

3.1.2 Step Graphon Parameterization

A step graphon approximates this continuous space as a piecewise-constant surface. We partition the unit interval $[0, 1]$ into M disjoint subsets $\{I_1, \ldots, I_M\}$. A probability vector $l \in \Delta^{M-1}$ defines the length of each subset such that $\lambda(I_i) = l_i$. The connection probability between subsets is defined by a blockwise affinity matrix $p \in [0, 1]^{M\times M}$. The step graphon is then:

$$W(x, y) = \sum_{i=1}^{M} \sum_{j=1}^{M} p_{ij}\, \mathbb{I}_{I_i}(x)\, \mathbb{I}_{I_j}(y) \tag{1}$$

where $\mathbb{I}$ is the indicator function.

3.1.3 Multifractal Connectivity via Kronecker Recursion

Standard step graphons implicitly assume non-hierarchical topologies. However, real-world networks exhibit varying levels of heterogeneity across different scales (Xue & Bogdan, 2017). This heterogeneity is formalized by the Lipschitz–Hölder exponent α, which measures how the local connectivity scales with resolution. A *monofractal* object has the same scaling behavior everywhere (a single α), while a *multifractal* object exhibits a distribution of scaling behaviors, characterized by the spectrum $f(\alpha)$ as its width ($\Delta\alpha = \alpha_{\max} - \alpha_{\min}$) and shape.

To capture this self-similar structure, we recursively expand the base parameters $\theta^{(1)} = (l^{(1)}, p^{(1)})$, where $l^{(1)} \in \text{int}(\Delta^{M-1})$ and $p^{(1)} \in (0, 1)^{M\times M}$, to a multiscale depth K via the Kronecker product:

$$l^{(K)} = \underbrace{l^{(1)} \otimes \cdots \otimes l^{(1)}}_{K \text{ times}}, \quad p^{(K)} = \underbrace{p^{(1)} \otimes \cdots \otimes p^{(1)}}_{K \text{ times}} \tag{2}$$

This recursive expansion partitions the space into M^K contiguous sub-intervals indexed by $q = (q_1, \ldots, q_K)$. The multifractal step graphon is defined as:

$$W_K(x, y) = \sum_{q} \sum_{h} p_{qh}^{(K)}\, \mathbb{I}_{I_q}(x)\, \mathbb{I}_{I_h}(y) \tag{3}$$

where the interaction amplitude factorizes across scales as $p_{qh}^{(K)} = \prod_{k=1}^{K} p_{q_k h_k}^{(1)}$. This multiplicative factorization enables varying levels of heterogeneity across scales: $\log p_{qh}^{(K)}$ is a sum of K terms from $\log p^{(1)}$, so different index paths yield different local scaling exponents. Uniform entries in $p^{(1)}$ collapse to a single exponent ($\Delta\alpha = 0$, monofractal), while heterogeneous entries widen the spectrum. Thus, $\theta^{(1)}$ compactly controls where W_K lies on the monofractal–multifractal continuum. The Kronecker structure also yields closed-form expressions for several network statistics, such as the multifractal spectrum, derived in Appendix B.

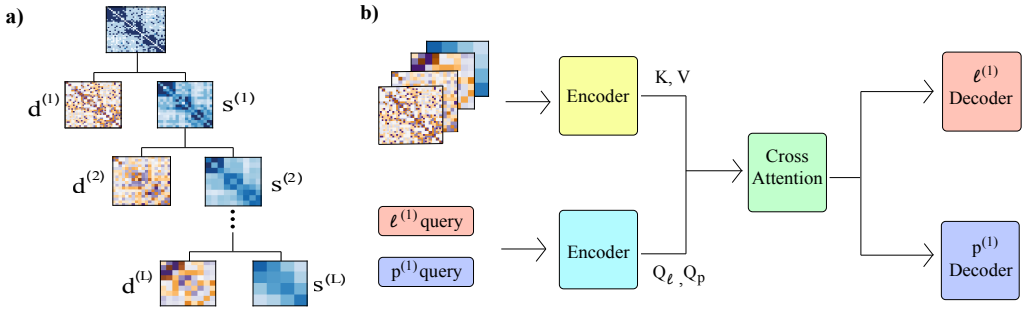

Figure 2: **Neural inverse operator architecture. (a)** The input adjacency matrix is decomposed across L scales by a 2D multiwavelet transform, yielding hierarchical coefficients that are tokenized and processed by a branch network (Keys and Values). **(b)** Learnable queries for $\ell^{(1)}$ and $p^{(1)}$ pass through a trunk network and are combined with the branch via cross-attention. Scale-specific outputs are decoded by MLPs to predict the base parameters.

3.1.4 Sampling Networks from the Multifractal Step Graphon

A finite graph $G = (V, E)$ is constructed by sampling from the multifractal step graphon W_K. By the Aldous representation, each node u is associated with a latent coordinate $x_u \sim \text{Uniform}(0, 1)$, which falls in the interval I_{q_u} with probability $l^{(K)}_{q_u}$. Because W_K is piecewise-constant on these intervals, this is equivalent to drawing community assignments directly:

$$q_u \sim \text{Multinomial}\big(l^{(K)}\big), \tag{4}$$

which is the form used in implementation. The continuous interaction intensity for pair (u, v) is then the constant value of W_K on the block $I_{q_u} \times I_{q_v}$:

$$W_{uv} = W_K(x_u, x_v) = p^{(K)}_{q_u q_v}. \tag{5}$$

To map this continuous amplitude to a discrete edge weight $r_{uv} \in \mathbb{N}_0$, we sample from a conditional geometric distribution:

$$P(r_{uv} = r \mid W_{uv}) = (1 - W_{uv})(W_{uv})^r. \tag{6}$$

Higher connection probabilities naturally produce heavier-tailed edge weights while maintaining support on all non-negative integers.

3.1.5 Identifiability

Inverse problems are typically ill-posed (Molinaro et al., 2023). We show identifiability (Proposition 1) of our formulation, which ensures that the inverse map is well-defined and distinct parameters produce distinct graphons.

Proposition 1 (Identifiability of the multifractal step graphon)**.** *Assume additionally that the expanded block profiles of $p^{(K)}$ are pairwise distinct, so that no two blocks have identical incoming and outgoing interaction profiles. Under this generic non-degeneracy condition, let $M \geq 2$ and $K \geq 2$. Consider two base parameterizations $\theta_1 = (l^{(1)}_1, p^{(1)}_1)$ and $\theta_2 = (l^{(1)}_2, p^{(1)}_2)$, where $l_i \in \text{int}(\Delta^{M-1})$ and $p_i \in (0, 1)^{M \times M}$. Let $W^{(1)}_K$ and $W^{(2)}_K$ denote the corresponding multifractal step graphons obtained via K-fold Kronecker expansion. If $W^{(1)}_K = W^{(2)}_K$ almost everywhere on $[0, 1]^2$, then there exists a permutation $\sigma \in S_M$ such that $l^{(1)}_1 = \sigma(l^{(1)}_2)$ and $p^{(1)}_1 = \sigma(p^{(1)}_2)$, where σ acts by reindexing: $\sigma(l)_i = l_{\sigma(i)}$ and $\sigma(p)_{ij} = p_{\sigma(i)\sigma(j)}$.*

The proof, given in Appendix C, proceeds in three steps: (i) the piecewise-constant graphon determines its block structure up to permutation via its discontinuities. (ii) the Kronecker factorization of the expanded parameters into base parameters is unique, by rank-1 symmetric tensor decomposition. (iii) the coupling between $l^{(K)}$ and $p^{(K)}$ through shared block indices forces the permutation to be consistent across both.

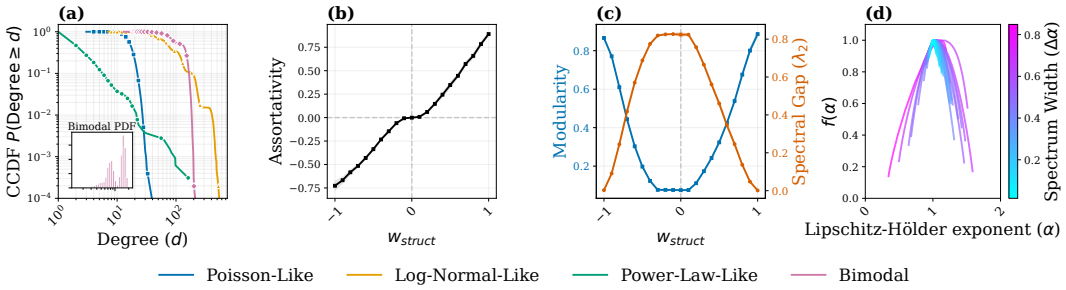

Figure 3: **Compact Parameterization of Diverse Topologies. (a)** Degree distributions span Poisson-like, log-normal-like, power-law-like (with exponential-tail), and bimodal regimes that are present in real-life networks. **(b)** Sweeping w_{struct} allows expression of varying levels of assortativity and disassortativity. **(c)** w_{struct} also controls the full range of modularity (left axis) and spectral gap λ_2 (right axis). **(d)** Multifractal spectra forms a continuous envelope from monofractal to multifractal.

4 Inference via Neural Operators

Recovering $\theta^{(1)} = (l^{(1)}, p^{(1)})$ from an observed graph requires learning the inverse operator $\mathcal{F} : W_K \mapsto \theta^{(1)}$. The observed adjacency matrix is treated as a finite-sample realization of a graph generated from $W_K \in L^2([0,1]^2)$. Because W_1 is fully parameterized by $\theta^{(1)}$, recovering these parameters fully specifies the multiscale graphon W_K.

Multifractal step graphons are low-regularity, piecewise-constant objects, for which multiwavelet bases are known to provide efficient representations (Alpert, 1993). We verify empirically that the multiwavelet basis resolves the recursive structure of W_K: the proportion of energy in the detail-coefficients grows exponentially with recursion depth (Appendix E). We instantiate the inverse operator as a Multiwavelet Neural Operator Transformer (MWNOT) (Fig.2), combining a 2D multiwavelet decomposition (Gupta et al., 2021; Xiao et al., 2023) with cross-attention for operator approximation (Shih et al., 2025). We use MWNOT as one architecture compatible with the proposed formulation, not as a claim of architectural optimality.

4.1 Architecture

First, a 2D discrete multiwavelet transform (Gupta et al., 2021) decomposes A across L spatial scales. The multiwavelet basis is constructed from shifted Legendre polynomials of order k and orthogonalized via Gram–Schmidt so that detail coefficients at each scale are orthogonal to the coarse approximation. At each scale ℓ, the transform yields coarse coefficients $s^{(\ell)}$ capturing macroscopic connectivity and detail coefficients $d^{(\ell)}$ capturing high-frequency variation.

Second, we adopt a branch–trunk cross-attention architecture (Lu et al., 2021; Shih et al., 2025) to map these multiscale coefficients to $\theta^{(1)}$. The branch network encodes the input function through its wavelet tokens, producing keys and values. The trunk network processes learnable query embeddings for $l^{(1)}$ and $p^{(1)}$. Cross-attention approximates the operator through a linear combination of input-derived features weighted by target-dependent queries (Shih et al., 2025).

Third, scale-specific attention outputs are concatenated across ℓ and decoded by MLPs to predict $\hat{l}^{(1)}$ and $\hat{p}^{(1)}$. Full architectural details are provided in Appendix D.

4.2 Training

Training details, including data generation, loss functions, optimization schedule, and hyperparameter settings, are provided in Appendix F.

5 Experiments

The central claim of this work is that the multifractal step graphon, paired with a neural inverse operator trained only on its synthetic realizations, recovers interpretable parameters of real-world

networks through amortized inference that generalizes across graph sizes. We support this claim through a series of experiments evaluating expressivity (Sec. 5.1), parameter recovery on unseen graph sizes (Sec. 5.2), and zero-shot transfer to empirical data of various domains and sizes (Sec. 5.3.1). Finally, we apply our framework to single-observation empirical networks (Sec. 5.3.2), where the inferred parameters reconstruct each network comparably to per-instance fitting and track a reversible change in brain state in a multi-subject EEG case study.

5.1 Compact Parameterization of Diverse Topologies

We empirically validate that the multifractal step graphon family covers various topologies observed in real-world networks. We sweep two axes of $\mathbf{p}^{(1)}$: a heterogeneity parameter $w_{\text{het}} \in [0, 1]$ controlling block-density variation, and a structure parameter $w_{\text{struct}} \in [-1, 1]$ interpolating from disassortative to assortative community organization. Sweep details are in Appendix G.

Figure 3 confirms that varying $\boldsymbol{\theta}^{(1)}$ alone produces networks with several qualitatively distinct regimes of (a) degree distributions (Broido & Clauset, 2019), (b) assortativity, (c) modularity with inverse relationship to spectral gap, and (d) multifractal spectrum width, spanning from monofractal to multifractal. Together, these results suggest that the model provides a broad synthetic pretraining distribution that may enable transfer to real-world networks in various domains.

5.2 Zero-Shot Transfer to Unseen Size

We next verify that the neural inverse operator recovers $\boldsymbol{\theta}^{(1)}$ accurately across the conditions encountered at deployment: varying graph sizes N and recursive depths K. A single MWNOT is trained on $N \in \{128, 512, 1024\}$ and $K \in \{2, 4, 6\}$, then evaluated on 100 held-out test samples per setting where performance is averaged across $K \in \{2, \ldots, 6\}$ for each N. We test our model's ability to generalize to unseen graph sizes. MAE for $\mathbf{p}^{(1)}$ and $\mathbf{l}^{(1)}$ remains stable as size is changed from $N = 128$ to $N = 8192$ (Fig. 4a). Consequently, our model allows the user to train on small graphs for inference on larger ones, which may enhance the broad applicability of the model. Recovery remains stable across hierarchical depths as well.

5.3 Zero-Shot Transfer to Empirical Data

5.3.1 Graph Generation Benchmark

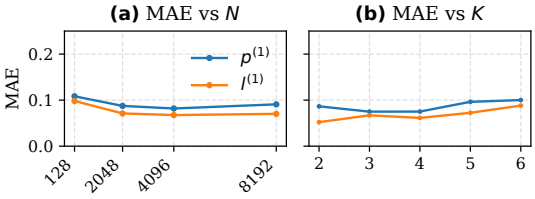

Figure 4: **Model Accuracy on Synthetic Graphs.** **(a)** The model generalizes to unseen graph sizes and **(b)** shows stable performance across varying levels of hierarchical structure.

The preceding sections show that the model spans diverse topologies (Sec. 5.1) and that its parameters are recovered accurately in graph sizes not seen during training (Sec. 5.2). We now test whether these combine to transfer to real-world graphs, using a model trained only on synthetic graphon realizations. We use the zero-shot graph-generation benchmark and baselines from Wang et al. (2025a). LGGM-X is pretrained on over 5,000 empirical graphs, with the target domain X held out. We use the same code, test sets, and metrics, and report baseline results from Wang et al. (2025a). The complete protocol is in Appendix H. As an ablation, **Ours-Mono** is trained on graphons with $p^{(1)} \equiv 0.5$, removing the topological diversity of the prior (Sec. 5.1).

Our model achieves the best average performance on three of four metrics ("ALL" in Table 1) and wins in all metrics in 4/12 domains. These results are notable given the contrasting training paradigms. LGGM-X is trained on thousands of real-world networks across many domains, whereas our model is trained *exclusively* on synthetic realizations of the multifractal step graphon. In addition, our method uses $\sim$22$\times$ fewer parameters ($\sim$205K vs. 4.61M). LGGM-X achieves stronger performance on some domains, such as infrastructure networks, where our method may struggle as it was trained on synthetic graphs without geometric priors. Averaged across domains, the ablation increases MMD on all four metrics, indicating that the topological diversity of the prior contributes to the transfer.

Table 1: Zero-shot Performance Comparison

Domain	Method	DEG	CC	Spec	Orb	Domain	Method	DEG	CC	Spec	Orb
FB	DiGress	0.3376	0.6298	0.0797	0.3593	BIO	DiGress	0.2712	0.5202	0.1127	0.3188
	LGGM-X	0.4723	0.6843	0.2924	0.7555		LGGM-X	0.1081	0.2696	0.0900	0.2053
	Ours-Mono	0.2782	0.3045	0.0308	0.0412		Ours-Mono	0.0879	0.1467	0.0827	0.1584
	Ours	**0.0589**	**0.0500**	**0.0284**	0.0403		**Ours**	**0.0442**	**0.0605**	**0.0629**	**0.1090**
ASN	DiGress	0.1496	0.3258	0.1506	0.4420	ECON	DiGress	0.2987	0.4841	0.2162	0.3834
	LGGM-X	**0.0281**	0.2440	**0.0830**	**0.0618**		LGGM-X	0.1213	0.0920	0.1120	0.1086
	Ours-Mono	0.1350	0.1375	0.1484	0.5197		Ours-Mono	0.1011	0.1092	0.0707	0.0916
	Ours	0.0780	0.1281	0.1003	0.3012		**Ours**	**0.0780**	**0.0474**	**0.0199**	**0.0598**
EMAIL	DiGress	0.2192	0.6012	0.0702	0.3416	RT	DiGress	0.4164	**0.1327**	0.4147	0.5957
	LGGM-X	0.0751	0.2364	0.0768	0.3089		LGGM-X	**0.0525**	0.1429	**0.1330**	0.2219
	Ours-Mono	0.1396	0.0921	0.0600	0.0765		Ours-Mono	0.2911	0.4537	0.3705	0.3612
	Ours	**0.0631**	**0.0529**	**0.0563**	**0.0725**		**Ours**	0.1999	0.4070	0.2345	0.2452
WEB	DiGress	0.2556	0.6186	0.1877	0.6045	COL	DiGress	0.2473	0.5826	0.2314	0.7679
	LGGM-X	**0.0648**	0.3961	**0.0549**	**0.1127**		LGGM-X	**0.0736**	0.5769	**0.0895**	**0.0988**
	Ours-Mono	0.1553	0.1456	0.1749	0.3319		Ours-Mono	0.2143	0.3022	0.2329	0.7161
	Ours	0.0692	**0.0795**	0.1145	0.2760		**Ours**	0.1020	**0.1683**	0.1863	0.6032
ROAD	DiGress	0.3705	0.8226	0.2801	0.7198	ECO	DiGress	0.5431	0.7915	0.2338	0.6045
	LGGM-X	**0.0713**	0.2193	**0.0987**	**0.2986**		LGGM-X	0.4753	0.3904	0.3194	0.3934
	Ours-Mono	0.4300	0.6884	0.3142	0.7433		Ours-Mono	0.1744	0.3025	0.0436	0.2044
	Ours	0.1107	**0.1850**	0.1446	0.6027		**Ours**	**0.0632**	**0.2298**	**0.0326**	0.2040
POWER	DiGress	0.3726	0.4582	0.3270	1.4732	CITATION	DiGress	0.2527	0.7790	0.1315	0.4966
	LGGM-X	**0.0119**	**0.1293**	**0.0373**	**0.0754**		LGGM-X	0.1348	0.7257	0.1160	0.4981
	Ours-Mono	0.1963	0.3947	0.2809	0.9536		Ours-Mono	0.1364	0.2654	0.0917	0.1601
	Ours	0.1660	0.2992	0.2265	0.6956		**Ours**	**0.0407**	**0.1150**	**0.0621**	**0.1010**
ALL	DiGress	0.3112	0.5622	0.2030	0.5923						
	LGGM-X	0.1408	0.3422	0.1253	**0.2616**						
	Ours-Mono	0.1950	0.2785	0.1584	0.3632						
	Ours	**0.0895**	**0.1519**	**0.1057**	0.2759						

DEG, CC, Spec, Orb: MMD of degree, clustering coefficient, Laplacian eigenvalues and orbit counts. Ours and Ours-Mono are means over 10 seeds. Full table with SDs in Table S1). On ECO, the orbit kernel saturates, giving the same value for any set of distinct dense samples. This cell is not used to mark the best method.

Taken together, the results suggest that the generative rule and the inference of its parameters transfer to real-world networks in various domains.

5.3.2 Applications to Single-Observation Empirical Data

Recent advances in connectomics have produced datasets with unprecedented scale, such as the complete connectome of *Drosophila* (Dorkenwald et al., 2024). However, these data often exist as a single observation. Whole-graph generation on these datasets has been inaccessible to existing deep generative models since they require training corpora, and foundation graph models are still in development (Wang et al., 2025b). In order to bridge this gap, we assess our model's ability to infer meaningful structures in such datasets, with sizes ranging from $N = 163$ to $N = 6301$.

Evaluation for single-graph datasets. Standard graph generation evaluation uses MMD between distributions of graph statistics computed across many graphs in test and generated sets (You et al., 2018; Vignac et al., 2022; Wang et al., 2025a). This metric is ill-defined at $N = 1$ since no test distribution exists with a single empirical graph. We instead compare intra-graph distributions between empirical and synthetic networks using 1-Wasserstein distances on degree distributions and Laplacian spectra, as well as mean squared error on the multifractal spectra (Appendix J).

Reference comparison with per-instance fitting. We calibrate against WMGM (Yang et al., 2021), which is to our knowledge the only generative-rule estimator for self-similar networks with weighted edges and node heterogeneity. WMGM optimizes on each network separately via EM. WMGM thus serves as a reference for what direct fitting achieves, not as a competing baseline. Our zero-shot method is better than or tied with WMGM on 10 of 18 comparisons (2 better, 8 tied, 8 worse), while running 9–550$\times$ faster (Table 2). Differences are largest in spectral error, where WMGM is better on four of six networks. This indicates that our zero-shot method reconstructs the multifractal and degree structure of real networks comparably to direct fitting, with a remaining gap in spectral structure.

Inferred parameters track brain state. To demonstrate the utility of our approach in scientific applications, we next ask whether the inferred parameters reflect reversible changes in network organization. We analyze alpha-band EEG connectivity from 20 subjects at baseline, under mild and moderate propofol sedation, and after recovery (Chennu et al., 2016) (Appendix K). We focus on the

Table 2: Reconstruction of Single-Instance Networks

Domain	Method	Time (ms)	MFrac	DegW	SpecW	Domain	Method	Time (ms)	MFrac	DegW	SpecW
C. elegans	WMGM	2060	2.68e-2	**5.67e-4**	**6.97e-2**	P2P Gnutella	WMGM	2540	5.19e-3	1.41e-4	**8.09e-2**
	Ours	**3.76**	**2.19e-4**	1.20e-3	1.27e-1		Ours	**198**	3.56e-3	1.23e-4	1.24e-1
Drosophila	WMGM	1930	7.35e-3	2.29e-4	**6.41e-2**	Power Grid	WMGM	1810	5.88e-4	4.23e-5	**1.81e-1**
	Ours	**14.1**	3.84e-3	2.69e-4	1.18e-1		Ours	**198**	1.24e-3	**2.54e-5**	2.58e-1
Allen Mouse	WMGM	1050	**1.52e-4**	1.38e-4	3.99e-3	MN Roads	WMGM	1490	**1.73e-3**	**5.92e-5**	1.86e-1
	Ours	**2.95**	7.86e-3	9.86e-5	6.82e-3		Ours	**50.9**	2.42e-3	1.24e-4	1.84e-1

MFrac: multifractal MSE; DegW: degree Wasserstein; SpecW: spectral Wasserstein. Error metrics are means over 10 seeds, with the full table with SD values reported in Table S2. Times are defined in Appendices F and I. Bold marks the lower value where the two methods' mean $\pm$ SD intervals do not overlap.

Table 3: Revealing Hidden Differences in EEG Networks Under Sedation.

Measure	Baseline	Mild	Moderate	Recovery	d Base − Mod	d Mild − Mod	d Rec − Mod	d Base − Rec
Inferred generative rule $\hat{\theta}^{(1)}$								
Assortativity index	-0.75 ± 0.32	-0.74 ± 0.27	-0.48 ± 0.34	-0.76 ± 0.29	$\mathbf{-0.84}$ (0.043)	$\mathbf{-0.87}$ (0.043)	$\mathbf{-0.91}$ (0.036)	0.03
$\Delta\alpha$	3.45 ± 2.29	3.44 ± 2.11	1.71 ± 1.86	3.55 ± 2.11	$\mathbf{0.83}$ (0.048)	$\mathbf{0.87}$ (0.019)	$\mathbf{0.92}$ (0.019)	-0.05
Classical network metrics								
Mean clustering	0.67 ± 0.12	0.66 ± 0.12	0.73 ± 0.16	0.68 ± 0.11	-0.40	-0.45	-0.36	-0.07
Mean shortest path	1.36 ± 0.16	1.37 ± 0.16	1.28 ± 0.16	1.36 ± 0.15	0.48	0.57	0.47	0.03
Modularity	0.053 ± 0.045	0.052 ± 0.030	0.041 ± 0.026	0.046 ± 0.031	0.32	0.39	0.19	0.16
Spectral gap	0.81 ± 0.12	0.81 ± 0.10	0.84 ± 0.08	0.82 ± 0.10	-0.31	-0.43	-0.23	-0.10

Values are mean $\pm$ SD over $n = 20$ sessions per state; the same 20 subjects contribute to every state. d is Cohen's d (pooled SD) for the first state minus the second. Bold marks values whose two-sided Mann–Whitney U test survives Holm correction over the four contrasts ($p < 0.05$). The corrected p is shown in parentheses only for significant values.

alpha band (8–15 Hz) functional network, where propofol produces its most prominent EEG changes on this frequency band (Purdon et al., 2013).

From $\hat{p}^{(1)}$ we derive two summaries, the assortativity index and multifractal width ($\Delta\alpha$) (Appendix K). Both metrics separate moderate sedation from every other state (Cohen's $|d| = 0.8$–0.9, the difference in means divided by the pooled SD, Holm-corrected $p < 0.05$), while values at baseline, mild sedation and recovery are similar (Table 3). Under moderate sedation the rule becomes less disassortative and its multifractal spectrum narrows ($\Delta\alpha$: $3.4 \rightarrow 1.7$), consistent with reported changes in multiscale complexity (Casali et al., 2013; Wilsenach et al., 2025) and integration (Luppi et al., 2021) under altered consciousness. Critically, both metrics return to levels similar to baseline on recovery, suggesting that they reflect a reversible, state-dependent change rather than drift over the recording session. In contrast, traditional network statistics show smaller effects ($|d| = 0.2$–0.6), none significant after correction. The two summaries also decode moderate sedation in unseen subjects (AUC 0.69–0.75), suggesting that they carry state information that is shared between subjects (Fig.S2A). Moreover, the fit does not differ across states, indicating that the difference in inferred parameters during moderate sedation is not confounded by changes in the model fit (Fig.S2B). Lastly, 76 of 80 networks fit within the error range expected when the model is true (Fig.S2C).

6 Discussion

Here we reiterate that comparisons with LGGM-X and WMGM serve as reference points for transfer and fit. As such, we do not claim that our formulation is intended to supersede deep or mechanistic models in general. Rather, it is a complementary method suited to settings requiring both interpretable parameters and amortized inference, such as in scientific applications when N=1 and networks are of sizes unseen during training. We also do not claim that MWNOT is the optimal inference method for this framework. Our approach has several limitations. For one, the model assumes the same connection rule repeats across scales. Thus, networks whose wiring rule varies irregularly across scales fall outside this assumption. In addition, the inverse operator is not permutation-equivariant. We resolve this with a deterministic degree-sorting canonicalization, which is always available without any metadata (Appendix E.1). Lastly, the model needs improvement in the spectral metrics. Despite these limitations, the results presented here indicate that mechanistic interpretability and amortized inference can be effectively combined within a unified framework to improve our understanding of complex networks. Future work includes making the generative rule more expressive and comparing other inverse-operator architectures to identify which yields the most accurate inference.

AI USE STATEMENT

In this work, we used generative AI tools to design and provide feedback on research methodology and experiments, implement methods, and interpret results. We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, or clean and reformat datasets, and generating synthetic datasets, assisting with translation, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to suggest experimental parameters, create and edit software code, create scientific figures, draft parts of the paper, edit the paper to improve readability, identify relevant literature, and format references. We have reviewed all AI-assisted work. All AI-generated code, analyses and text edits were reviewed by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We describe the full experimental protocol in the Appendix, including preprocessing, training settings, hyperparameters, and evaluation procedures. Anonymized source code to reproduce all experiments is provided in the supplementary materials, with instructions for running each experiment.

REFERENCES

Dimitris Achlioptas, Aaron Clauset, David Kempe, and Cristopher Moore. On the bias of traceroute sampling: or, power-law degree distributions in regular graphs. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pp. 694–703, 2005.

Edo M Airoldi, Thiago B Costa, and Stanley H Chan. Stochastic blockmodel approximation of a graphon: Theory and consistent estimation. *Advances in Neural Information Processing Systems*, 26, 2013.

Danyal Akarca, Petra E Vértes, Edward T Bullmore, and Duncan E Astle. A generative network model of neurodevelopmental diversity in structural brain organization. *Nature communications*, 12(1):4216, 2021.

David J Aldous. Representations for partially exchangeable arrays of random variables. *Journal of Multivariate Analysis*, 11(4):581–598, 1981.

Bradley K Alpert. A class of bases in L^2 for the sparse representation of integral operators. *SIAM journal on Mathematical Analysis*, 24(1):246–262, 1993.

Ali Azizpour, Nicolas Zilberstein, and Santiago Segarra. Scalable implicit graphon learning. *arXiv preprint arXiv:2410.17464*, 2024.

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

Michael Batty and Paul A Longley. *Fractal cities: a geometry of form and function*. Academic press, 1994.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. doi: 10.1080/01621459.2017.1285773.

Christian Borgs, Jennifer T Chayes, László Lovász, Vera T Sós, and Katalin Vesztergombi. Convergent sequences of dense graphs i: Subgraph frequencies, metric properties and testing. *Advances in Mathematics*, 219(6):1801–1851, 2008.

Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Gorke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On modularity clustering. *IEEE transactions on knowledge and data engineering*, 20(2):172–188, 2008.

Anna D Broido and Aaron Clauset. Scale-free networks are rare. *Nature communications*, 10(1): 1017, 2019.

Adenauer G Casali, Olivia Gosseries, Mario Rosanova, Mélanie Boly, Simone Sarasso, Karina R Casali, Silvia Casarotto, Marie-Aurélie Bruno, Steven Laureys, Giulio Tononi, and Marcello Massimini. A theoretically based index of consciousness independent of sensory processing and behavior. *Science Translational Medicine*, 5(198):198ra105–198ra105, 2013.

Stanley Chan and Edoardo Airoldi. A consistent histogram estimator for exchangeable graph models. In *International Conference on Machine Learning*, pp. 208–216. PMLR, 2014.

Shaw-Ji Chen, Ding-Lieh Liao, Chia-Hsiang Chen, Tse-Yi Wang, and Kuang-Chi Chen. Construction and analysis of protein-protein interaction network of heroin use disorder. *Scientific reports*, 9(1): 4980, 2019.

Srivas Chennu, Stuart O'Connor, Ram Adapa, David K Menon, and Tristan A Bekinschtein. Brain connectivity dissociates responsiveness from drug exposure during propofol-induced transitions of consciousness. *PLoS computational biology*, 12(1):e1004669, 2016.

J-J Daudin, Franck Picard, and Stéphane Robin. A mixture model for random graphs. *Statistics and computing*, 18(2):173–183, 2008.

Sven Dorkenwald, Arie Matsliah, Amy R Sterling, Philipp Schlegel, Szi-Chieh Yu, Claire E McKellar, Albert Lin, Marta Costa, Katharina Eichler, Yijie Yin, et al. Neuronal wiring diagram of an adult brain. *Nature*, 634(8032):124–138, 2024.

Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

Gaurav Gupta, Xiongye Xiao, and Paul Bogdan. Multiwavelet-based operator learning for differential equations. *Advances in neural information processing systems*, 34:24048–24062, 2021.

Bruce Hajek. Cooling schedules for optimal annealing. *Mathematics of operations research*, 13(2): 311–329, 1988.

Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. Gnot: A general neural operator transformer for operator learning. In *International conference on machine learning*, pp. 12556–12569. PMLR, 2023.

Thomas N Kipf and Max Welling. Variational graph auto-encoders. In *NeurIPS Workshop on Bayesian Deep Learning*, 2016.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3): 455–500, 2009.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

Daria La Rocca, Nicolas Zilber, Patrice Abry, Virginie van Wassenhove, and Philippe Ciuciu. Self-similarity and multifractality in human brain activity: A wavelet-based analysis of scale-free brain dynamics. *Journal of neuroscience methods*, 309:175–187, 2018.

Chang-Yong Lee and Sunghwan Jung. Statistical self-similar properties of complex networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 73(6):066102, 2006.

Sang Hoon Lee, Pan-Jun Kim, and Hawoong Jeong. Statistical properties of sampled networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 73(1):016102, 2006.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: an approach to modeling networks. *Journal of Machine Learning Research*, 11 (2), 2010.

Yujia Li, Oriol Vinyals, Chris Dyer, Razvan Pascanu, and Peter Battaglia. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

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. *Advances in neural information processing systems*, 32, 2019.

Jin-Long Liu, Jian Wang, Zu-Guo Yu, and Xian-Hua Xie. Fractal and multifractal analyses of bipartite networks. *Scientific reports*, 7(1):45588, 2017.

Yuqing Long and Yanguang Chen. Multifractal scaling analyses of urban street network structure: The cases of twelve megacities in china. *PLoS One*, 16(2):e0246925, 2021.

László Lovász and Balázs Szegedy. Limits of dense graph sequences. *Journal of Combinatorial Theory, Series B*, 96(6):933–957, 2006.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via deeponet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

Andrea I Luppi, Michael M Craig, Ioannis Pappas, Paola Finoia, Guy B Williams, Judith Allanson, John D Pickard, Adrian M Owen, Lorina Naci, David K Menon, and Emmanuel A Stamatakis. Brain network integration dynamics are associated with loss and recovery of consciousness induced by sevoflurane. *Human Brain Mapping*, 42(9):2802–2822, 2021.

Roberto Molinaro, Yunan Yang, Björn Engquist, and Siddhartha Mishra. Neural inverse operators for solving pde inverse problems. *arXiv preprint arXiv:2301.11167*, 2023.

Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling. In *International conference on artificial intelligence and statistics*, pp. 4474–4484. PMLR, 2020.

Gergely Palla, László Lovász, and Tamás Vicsek. Multifractal network generator. *Proceedings of the National Academy of Sciences*, 107(17):7640–7645, 2010.

Patrick L Purdon, Eric T Pierce, Eran A Mukamel, Michael J Prerau, John L Walsh, Kin Foon K Wong, Andres F Salazar-Gomez, Priscilla G Harrell, Aaron L Sampson, Aylin Cimenser, et al. Electroencephalogram signatures of loss and recovery of consciousness from propofol. *Proceedings of the National Academy of Sciences*, 110(12):E1142–E1151, 2013.

Benjamin Shih, Ahmad Peyvan, Zhongqiang Zhang, and George Em Karniadakis. Transformers as neural operators for solutions of differential equations with finite regularity. *Computer Methods in Applied Mechanics and Engineering*, 434:117560, 2025.

Martin Simonovsky and Nikos Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. In *International conference on artificial neural networks*, pp. 412–422. Springer, 2018.

Michael PH Stumpf, Carsten Wiuf, and Robert M May. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proceedings of the National Academy of Sciences*, 102(12):4221–4224, 2005.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. *arXiv preprint arXiv:2209.14734*, 2022.

Martin Vinck, Robert Oostenveld, Marijn Van Wingerden, Franscesco Battaglia, and Cyriel MA Pennartz. An improved index of phase-synchronization for electrophysiological data in the presence of volume-conduction, noise and sample-size bias. *Neuroimage*, 55(4):1548–1565, 2011.

Yu Wang, Ryan A Rossi, Namyong Park, Huiyuan Chen, Nesreen K Ahmed, Puja Trivedi, Franck Dernoncourt, Danai Koutra, and Tyler Derr. A large-scale training paradigm for graph generative models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Zehong Wang, Zheyuan Liu, Tianyi Ma, Jiazheng Li, Zheyuan Zhang, Xingbo Fu, Yiyang Li, Zhengqing Yuan, Wei Song, Yijun Ma, et al. Graph foundation models: A comprehensive survey. *arXiv preprint arXiv:2505.15116*, 2025b.

Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393 (6684):440–442, 1998.

James Barnard Wilsenach, Charlotte M Deane, Gesine Reinert, and Katie Warnaby. Graph models of brain state in deep anesthesia reveal sink state dynamics of reduced spatiotemporal complexity. *Network Neuroscience*, 9(4):1176–1198, 2025.

C. F. Jeff Wu. On the convergence properties of the em algorithm. *The Annals of Statistics*, 11(1): 95–103, 1983. doi: 10.1214/aos/1176346060.

Xinyue Xia, Gal Mishne, and Yusu Wang. Implicit graphon neural representation. In *International Conference on Artificial Intelligence and Statistics*, pp. 10619–10634. PMLR, 2023.

Xiongye Xiao, Defu Cao, Ruochen Yang, Gaurav Gupta, Gengshuo Liu, Chenzhong Yin, Radu Balan, and Paul Bogdan. Coupled multiwavelet neural operator learning for coupled partial differential equations. *arXiv preprint arXiv:2303.02304*, 2023.

Hongteng Xu, Peilin Zhao, Junzhou Huang, and Dixin Luo. Learning graphon autoencoders for generative graph modeling. *arXiv preprint arXiv:2105.14244*, 2021.

Yuankun Xue and Paul Bogdan. Reliable multi-fractal characterization of weighted complex networks: algorithms and implications. *Scientific reports*, 7(1):7487, 2017.

Ruochen Yang and Paul Bogdan. Controlling the multifractal generating measures of complex networks. *Scientific reports*, 10(1):5541, 2020.

Ruochen Yang, Frederic Sala, and Paul Bogdan. Hidden network generating rules from partially observed complex networks. *Communications Physics*, 4(1):199, 2021.

Fang-Cheng Yeh, Sandip Panesar, David Fernandes, Antonio Meola, Masanori Yoshino, Juan C Fernandez-Miranda, Jean M Vettel, and Timothy Verstynen. Population-averaged atlas of the macroscale human structural connectome and its network topology. *Neuroimage*, 178:57–68, 2018.

Chenzhong Yin, Xiongye Xiao, Valeriu Balaban, Mikhail E Kandel, Young Jae Lee, Gabriel Popescu, and Paul Bogdan. Network science characteristics of brain-derived neuronal cultures deciphered from quantitative phase imaging data. *Scientific reports*, 10(1):15078, 2020.

Jiaxuan You, Rex Ying, Xiang Ren, William Hamilton, and Jure Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models. In *International conference on machine learning*, pp. 5708–5717. PMLR, 2018.

Supplemental Material

A Model Selection

Selecting M. M is the number of base communities. The base graphon splits the latent space $[0, 1]$ into M intervals and each recursion step splits every interval into M sub-intervals. The depth-K graphon therefore partitions the latent space into M^K blocks, and each block must contain enough nodes for its parameters to be reflected in the observed graph. For our smallest training graphs at the largest depth ($N = 512$, $K = 6$; training pairs $K = 6$ only with $N \geq 512$), $M = 2$ yields $2^6 = 64$ blocks, whereas $M = 4$ would yield $4^6 = 4096$ blocks, far exceeding the number of nodes and leaving most blocks empty. Supporting larger M would therefore require exponentially larger training graphs.

Expressivity. Fixing $M = 2$ does not restrict the model to two communities. Expanding two recursive steps, $l^{(1)} \otimes l^{(1)}$ and $p^{(1)} \otimes p^{(1)}$ form a valid $M = 4$ base rule, so an $M = 2$ model at depth K is equivalent to the subset of the $M = 4$ model class at depth $K/2$ whose base rule is Kronecker-factorizable. What an unrestricted $M = 4$ model adds is capacity for non-factorizable branching rules. Periodic multirule generators (Appendix L) offer intermediate expressivity without requiring larger training graphs, and are a natural direction for future work.

B Closed-Form Network Statistics under the Multifractal Step Graphon

A consequence of the Kronecker structure of W_K is that several network statistics admit closed-form expressions that factorize across scales. We derive two such expressions here: the expected edge density and the expected degree of a node assigned to a given block.

B.1 Expected edge density

The expected edge density of W_K is

$$\rho(W_K) = \int_0^1 \int_0^1 W_K(x, y)\, dx\, dy = \sum_{q,h} l_q^{(K)}\, l_h^{(K)}\, p_{qh}^{(K)}. \tag{S1}$$

Because $l^{(K)}$ and $p^{(K)}$ each factorize as K-fold Kronecker products, the sum decomposes into a product of K identical scale-1 sums:

$$\rho(W_K) = \prod_{k=1}^{K} \left(\sum_{i,j} l_i^{(1)}\, l_j^{(1)}\, p_{ij}^{(1)} \right) = \rho(W_1)^K. \tag{S2}$$

Edge density at depth K is therefore the K-th power of the depth-1 density, making explicit the recursive nature of the construction.

B.2 Expected degree of a node in block q

For a graph of N nodes with latent coordinates drawn iid from Uniform$(0, 1)$, the expected out-degree of a node assigned to block $q = (q_1, \ldots, q_K)$ is

$$\mathbb{E}[d_q^{\text{out}}] = N \sum_h l_h^{(K)}\, p_{qh}^{(K)} = N \prod_{k=1}^{K} \left(\sum_j l_j^{(1)}\, p_{q_k j}^{(1)} \right). \tag{S3}$$

The expression for in-degree is analogous, with $p_{q_k j}^{(1)}$ replaced by $p_{j q_k}^{(1)}$. Each scale-k factor depends only on q_k, so the expected degree of a node decomposes into independent contributions from each level of the recursion.

B.3 MULTIFRACTAL SPECTRUM

The multifractal structure of W_K admits a closed-form characterization through the partition function. Define the scale-1 moment sum

$$Z(q) = \sum_{i,j} l_i^{(1)}\, l_j^{(1)} \left(p_{ij}^{(1)}\right)^q. \tag{S4}$$

Because the multiplicative structure of W_K makes the depth-K partition function factorize as $Z_K(q) = Z(q)^K$, the mass exponent $\tau(q)$ takes the closed form

$$\tau(q) = -\frac{\log Z(q)}{\log M}. \tag{S5}$$

The Lipschitz–Hölder exponent $\alpha(q)$ and multifractal spectrum $f(\alpha)$ are then obtained via the Legendre transform:

$$\alpha(q) = \frac{d\tau}{dq}, \qquad f(\alpha(q)) = q\,\alpha(q) - \tau(q). \tag{S6}$$

The full multifractal spectrum is therefore determined by $\theta^{(1)}$ alone, independent of K in the limit. This closed form makes precise the claim in Section 3.1.3 that $\theta^{(1)}$ controls where W_K lies on the monofractal–multifractal continuum: monofractal behavior corresponds to $Z(q) = c^q$ for some constant c, in which case $\tau(q)$ is linear and $f(\alpha)$ collapses to a single point; heterogeneity in $p^{(1)}$ produces curvature in $\tau(q)$ and a non-degenerate spectrum.

B.4 IMPLICATION

These factorizations show that macroscopic network statistics inherit the recursive structure of the generator: depth-K observables are powers or products of depth-1 quantities, and the multifractal spectrum is fully characterized by $\theta^{(1)}$ alone. This provides analytical tools for relating $\theta^{(1)}$ to observable network properties, and confirms that the recursive construction is not merely a notational device but produces statistics that scale predictably with depth.

C PROOF OF PROPOSITION 1 (IDENTIFIABILITY)

We establish identifiability through three steps: (i) the step graphon W_K determines its block structure uniquely up to permutation, (ii) the Kronecker factorization of the block parameters is unique, and (iii) the combined result yields identifiability of the base parameters up to community label permutation.

C.1 STEP 1: A STEP GRAPHON DETERMINES ITS BLOCK PARAMETERS UP TO PERMUTATION

A step graphon W_K is a piecewise-constant function on $[0,1]^2$ defined by a partition of $[0,1]$ into M^K contiguous intervals with lengths $l^{(K)} \in \mathrm{int}(\Delta^{M^K-1})$ and a block interaction matrix $p^{(K)} \in (0,1)^{M^K \times M^K}$. We show that these parameters are determined by W_K up to a simultaneous permutation of the blocks.

Since all entries of $l^{(K)}$ are strictly positive, the partition consists of M^K intervals each with positive Lebesgue measure. The function W_K takes the constant value $p_{qh}^{(K)}$ on the rectangle $I_q \times I_h$.

Suppose two parameterizations $(l_1^{(K)}, p_1^{(K)})$ and $(l_2^{(K)}, p_2^{(K)})$ yield the same graphon W_K almost everywhere. Because the graphon is piecewise constant with finitely many pieces, equality a.e. implies equality on every rectangle of positive measure.

The partition boundaries are determined by the discontinuities of W_K. For a fixed y in the interior of some interval I_h, the function $W_K(\cdot, y)$ is piecewise constant in x, taking the value $p_{qh}^{(K)}$ when $x \in I_q$. The discontinuities of this function occur at the boundaries between intervals I_q and I_{q+1} whenever $p_{qh}^{(K)} \neq p_{(q+1)h}^{(K)}$. By varying y across all intervals, a boundary between I_q and I_{q+1} is undetectable only if $p_{qh}^{(K)} = p_{(q+1)h}^{(K)}$ for every h — that is, blocks q and $q+1$ have identical interaction profiles with all other blocks.

In this case, blocks q and $q+1$ are indistinguishable: they have the same size (since both parameterizations must agree on the set of interval widths) and the same interaction profile. Swapping their labels produces the same graphon. This is precisely a permutation of the block indices.

Because the intervals are contiguous (ordered left-to-right on $[0,1]$), the set of partition boundaries is a property of the function itself, not of the parameterization. Both parameterizations must use the same boundary points and therefore the same set of interval widths. The only freedom is in which interval is assigned which label. Thus, there exists a permutation $\pi \in S_{M^K}$ such that:

$$l^{(K)}_{1,q} = l^{(K)}_{2,\pi(q)}, \quad p^{(K)}_{1,qh} = p^{(K)}_{2,\pi(q)\pi(h)} \tag{S7}$$

for all $q, h \in \{1, \dots, M^K\}$. □

C.2 Step 2: Uniqueness of the Kronecker factorization

We now show that if two base parameterizations (l_1, p_1) and (l_2, p_2) yield the same expanded parameters under K-fold Kronecker product (up to a permutation $\pi \in S_{M^K}$), then they are related by a base-level permutation $\sigma \in S_M$.

C.2.1 Step 2a: Factorization of the length vector

By construction, $l^{(K)} = l^{(1)} \otimes \cdots \otimes l^{(1)}$ (K times). The entries of $l^{(K)}$ are indexed by multi-indices $q = (q_1, \dots, q_K) \in \{1, \dots, M\}^K$ and take the form:

$$l^{(K)}_q = \prod_{k=1}^{K} l^{(1)}_{q_k} \tag{S8}$$

These are the entries of a rank-1 symmetric tensor $l^{(1)} \otimes \cdots \otimes l^{(1)} \in \mathbb{R}^{M \times \cdots \times M}$. We show that $l^{(1)}$ is recoverable from this tensor up to permutation of its entries.

Consider the $M \times M^{K-1}$ matricization of $l^{(K)}$, denoted $L_{(1)}$, obtained by unfolding along the first mode:

$$L_{(1)} = l^{(1)} \cdot \left(l^{(1)} \otimes \cdots \otimes l^{(1)} \right)^\top \tag{S9}$$

Since all entries of $l^{(1)}$ are strictly positive, $L_{(1)}$ has rank 1 and its column space is spanned by $l^{(1)}$. The simplex constraint $\sum_i l^{(1)}_i = 1$ fixes the scale uniquely. The only remaining ambiguity is permutation of the entries of $l^{(1)}$.

To see why the permutation must act consistently across all modes: a permutation $\pi \in S_{M^K}$ of the multi-indices that maps one Kronecker product to another must act independently on each coordinate of the multi-index. If π maps $(q_1, \dots, q_K)$ such that the first coordinate q_1 is permuted by σ in one multi-index but left unchanged in another, the multiplicative structure $l^{(K)}_q = \prod_k l^{(1)}_{q_k}$ would be violated. Formally, the logarithms of the entries form an additive rank-1 tensor:

$$\log l^{(K)}_q = \sum_{k=1}^{K} \log l^{(1)}_{q_k} \tag{S10}$$

A permutation preserving this additive structure must act as the same base permutation σ on each coordinate independently (Kolda & Bader, 2009).

Therefore, if $l^{(K)}_1 = \pi(l^{(K)}_2)$, then π is induced by a base permutation $\sigma \in S_M$ acting as $\sigma(q_1, \dots, q_K) = (\sigma(q_1), \dots, \sigma(q_K))$, and $l_1 = \sigma(l_2)$.

C.2.2 Step 2b: Factorization of the interaction matrix

Similarly, $p^{(K)} = p^{(1)} \otimes \cdots \otimes p^{(1)}$ (K times). The entries satisfy:

$$p^{(K)}_{qh} = \prod_{k=1}^{K} p^{(1)}_{q_k h_k} \tag{S11}$$

Let $\mathrm{vec}(p^{(1)}) \in (0,1)^{M^2}$ denote the vectorization of $p^{(1)}$. Then:

$$\mathrm{vec}(p^{(K)}) = \mathrm{vec}(p^{(1)}) \otimes \cdots \otimes \mathrm{vec}(p^{(1)}) \tag{S12}$$

By the same rank-1 tensor uniqueness argument as in Step 2a, $\mathrm{vec}(p^{(1)})$ is recoverable up to permutation. However, this permutation is not arbitrary over M^2 entries — it must be consistent with the base-level permutation σ established in Step 2a.

The permutation π acts simultaneously on both $l^{(K)}$ and $p^{(K)}$, because both are defined over the same block indices. Block q has size $l_q^{(K)}$ and interaction values $p_{qh}^{(K)}$. Any permutation that preserves both must be factorizable as σ acting independently on each level of the multi-index. The permutation σ from Step 2a induces the action:

$$p_{1,ij} = p_{2,\sigma(i)\sigma(j)} \tag{S13}$$

Therefore $p_1 = \sigma(p_2)$ under the same permutation σ. □

C.3 Step 3: Combining the results

From Step 1, $W_K^{(1)} = W_K^{(2)}$ a.e. implies that $(l_1^{(K)}, p_1^{(K)})$ and $(l_2^{(K)}, p_2^{(K)})$ agree up to a block permutation $\pi \in S_{M^K}$. From Step 2, the Kronecker structure forces π to be induced by a base-level permutation $\sigma \in S_M$, yielding $l_1 = \sigma(l_2)$ and $p_1 = \sigma(p_2)$.

This establishes that the multifractal step graphon W_K uniquely determines the base parameters $\theta^{(1)} = (l^{(1)}, p^{(1)})$ up to community label permutation.

C.4 Discussion of Assumptions

Strict positivity of $l^{(1)}$. The requirement $l^{(1)} \in \mathrm{int}(\Delta^{M-1})$ ensures that every community has positive measure and contributes to the graphon. If some $l_i^{(1)} = 0$, the corresponding community is absent and the effective model has fewer than M communities, reducing to an identifiable model of smaller dimension.

Strict positivity of $p^{(1)}$. The requirement $p_{ij}^{(1)} \in (0,1)$ ensures that the vectorized interaction matrix has no zero entries, which is needed for the rank-1 tensor factorization to be unique. If $p_{ij}^{(1)} = 0$ for some (i,j), multiple factorizations may produce the same zero entry. In practice, exact zeros are measure-zero in the parameter space and do not arise in the sampling procedure described in Appendix F.

$K \geq 2$. The result requires at least two Kronecker expansions. For $K = 1$, the graphon is a standard step graphon with M blocks, and identifiability reduces to the classical step graphon case (covered by Step 1 alone).

Note this proposition does not claim identifiability over arbitrary graphon representatives under all measure-preserving transformations, nor does it imply statistical stability from finite noisy graph samples.

D Multiwavelet Neural Operator Transformer

D.1 Architecture

Given an observed adjacency matrix $A \in \mathbb{R}^{N \times N}$, the model processes the graph topology through the following operations.

Channel lift. A learned linear map lifts each entry of the input A to k^2 channels, reshaping it into $x^{(0)} \in \mathbb{R}^{N \times N \times k^2}$. The Legendre polynomial basis enters through the fixed multiwavelet filters.

Multiwavelet decomposition. A 2D discrete multiwavelet transform is applied over L spatial scales using a non-standard wavelet representation (Gupta et al., 2021). The multiwavelet basis is

constructed via Gram–Schmidt orthogonalization of shifted Legendre polynomials, ensuring that high-frequency details are strictly orthogonal to the coarse approximation. At each scale $\ell \in \{1, \dots, L\}$, the feature grid is decomposed by a coarse filter $\mathcal{H}$ and a composite detail filter $\mathcal{G}$:

$$s^{(\ell)} = x^{(\ell-1)} * \mathcal{H} \tag{S14}$$

$$d^{(\ell)} = x^{(\ell-1)} * \mathcal{G} \tag{S15}$$

The coarse coefficients $s^{(\ell)}$ capture macroscopic connectivity, while $d^{(\ell)}$ concatenates horizontal (LH), vertical (HL), and diagonal (HH) high-frequency variations. The output $x^{(\ell)} = s^{(\ell)}$ serves as input for the next coarser scale. This hierarchical decomposition mirrors the multiplicative construction of the multifractal step graphon. When N is not a power of two, the canonicalised input (Appendix E.1) is cropped to the largest dyadic size $\leq N$, which drops the lowest-degree nodes.

Multiscale tokenization. The wavelet coefficients (details $d^{(\ell)}$ for $\ell \leq L$ and the final coarse $s^{(L)}$) are flattened into sequences and concatenated with positional grids and scale-indicator labels to form multiscale tokens $T^{(\ell)}$.

Cross-attention for operator approximation. The model parameterizes the inverse operator via cross-attention in a branch-trunk architecture (Lu et al., 2021; Shih et al., 2025; Hao et al., 2023). The multiscale tokens encode the input function (branch), while learnable query embeddings evaluate the target parameters (trunk).

Let $Q_l, Q_p \in \mathbb{R}^{M \times D}$ denote query matrices for $l^{(1)}$ and $p^{(1)}$. At each scale ℓ, tokens $T^{(\ell)}$ are projected into Keys $K^{(\ell)} \in \mathbb{R}^{N_\ell \times D}$ and Values $V^{(\ell)} \in \mathbb{R}^{N_\ell \times S}$. The cross-attention aggregates these into fixed-size latent representations:

$$z_s^{(\ell)}(m) = \sum_{d=1}^{D} \left(\frac{1}{N_\ell} \sum_{j=1}^{N_\ell} k_d \left(w_j^{(\ell)} \right)^T v_s \left(w_j^{(\ell)} \right) \right) q_d(m) \tag{S16}$$

where $m \in \{1, \dots, M\}$ indexes the target community and $w_j^{(\ell)}$ is the j-th wavelet token within $T^{(\ell)}$. The inner summation averages over N_ℓ grid locations, forming a data-dependent basis from the wavelet coefficients. The query embedding $q_d(m)$ extracts the relevant features, yielding attention outputs $Z_l^{(\ell)}, Z_p^{(\ell)} \in \mathbb{R}^{M \times S}$.

Decoding. Scale-specific representations are concatenated into global embeddings $Z_p = [Z_p^{(1)}, \dots, Z_p^{(L)}]$ and $Z_l = [Z_l^{(1)}, \dots, Z_l^{(L)}]$. MLPs decode these to predict the base parameters:

$$\hat{p}^{(1)} = \text{MLP}_p(Z_p) \in [0,1]^{M \times M} \tag{S17}$$

$$\hat{l}^{(1)} = \text{MLP}_l(Z_l) \in \Delta^{M-1} \tag{S18}$$

E Empirical Validation of the Multiwavelet Basis

We empirically verify that the multiwavelet basis used in MWNOT is responsive to the recursive multiscale construction of the multifractal step graphon to support the inductive bias of using this basis for multifractal objects.

Method. For each recursion depth $K \in \{1, \dots, 6\}$, we construct the continuous graphon W_K on an $N \times N$ grid ($N = 1024$) using fixed base parameters $l^{(1)} = [0.4, 0.6]$ and $p^{(1)} = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$ for demonstration. We project W_K onto the constant component of the Legendre multiwavelet basis and recursively apply the 2D multiwavelet transform L times. At each scale ℓ, we record the squared-norm energy of the detail coefficients $E^{(\ell)} = \sum |d^{(\ell)}|^2$, and after the final iteration, we record the energy of the coarse approximation $E^{(L)}_{\text{coarse}} = \sum |s^{(L)}|^2$. From these we compute two summary quantities:

$$\rho_{\text{fine}}(K) = \frac{E^{(1)} + E^{(2)}}{E^{(L)}_{\text{coarse}} + \sum_{\ell=1}^{L} E^{(\ell)}}, \tag{S19}$$

$$\rho_{\text{detail}}(K) = \frac{\sum_{\ell=1}^{L} E^{(\ell)}}{E^{(L)}_{\text{coarse}} + \sum_{\ell=1}^{L} E^{(\ell)}}. \tag{S20}$$

Result. Figure S1 reports both quantities as a function of K. Both ρ_{fine} and ρ_{detail} grow approximately exponentially with K (linear on a semi-log scale), spanning roughly two orders of magnitude over the recursion depths used in training.

Interpretation. Each recursion level introduces finer-scale block boundaries to W_K, which the multiwavelet basis encodes through its detail coefficients. Deeper recursion therefore produces an exponentially larger share of energy in the detail components. MWNOT's encoder and cross-attention operate on these detail coefficients; the architecture's representational capacity is thus allocated to the components where the multifractal structure of W_K resides. We interpret this as an empirical match between the basis and the construction, supporting the multiwavelet basis as a principled choice for representing this object class.

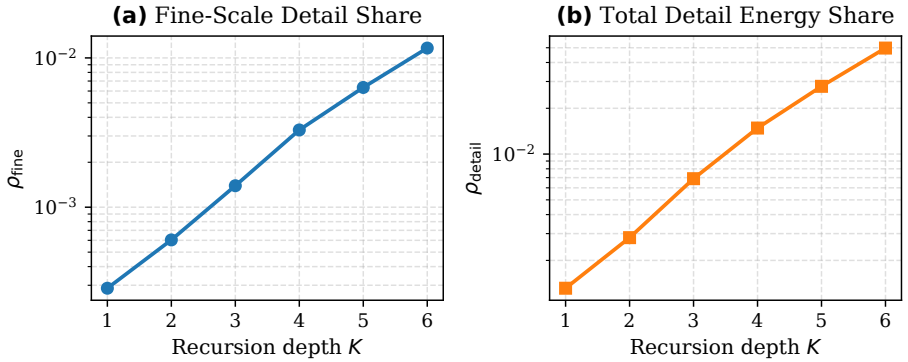

Figure S1: **Multiwavelet energy response to recursion depth.** (a) Fine-scale detail share ρ_{fine}. (b) Total detail energy share ρ_{detail}. Both quantities grow approximately exponentially with K, consistent with the multiplicative scaling structure of the multifractal step graphon.

E.1 Input Ordering

Function-space methods operate on inputs aligned to spatial axes. More specifically, the multiwavelet basis decomposes the adjacency matrix as a discretized function on $[0, 1]^2$, which benefits from inputs whose block structure is aligned with the spatial axes. As such, function-space formulations trade architectural permutation equivariance for a parameterization that allows for generalization to unseen graph sizes at inference.

This consideration concerns only the finite-sample to graphon inverse map learned by the architecture. It is distinct from the population-level identifiability of $W_k \mapsto \theta^{(1)}$ established in Proposition 1, which holds at the level of graphons, and from the exchangeability of finite graphs sampled from a graphon under the Aldous framework, which guarantees that *generation* of finite samples under our model remains permutation-invariant.

Canonicalization. We resolve the ordering dependence with a single deterministic, parameter-free rule applied to every input, synthetic or empirical: nodes are sorted in descending order of degree (weighted strength for weighted inputs). Ties are broken by a stable sort in the zero-shot benchmark and arbitrarily elsewhere. The rule uses no metadata, is available for any graph, and makes the model input a function of the graph alone: two labelings of the same graph yield the same sorted adjacency matrix up to the order of equal-degree nodes, so the inferred $\hat{\theta}^{(1)}$ is invariant to node relabeling up to that residual tie ambiguity. Degree sorting also recovers an approximately block-aligned form for the wavelet basis, since expected degree is monotone in community membership under the model (Appendix B).

Training. Synthetic graphs are produced by multinomial community assignment followed by geometric edge sampling (Appendix F). This randomizes which node sits in which community, producing adjacency matrices that resemble static noise rather than block-structured signals; the canonicalization above is applied to every generated training graph before augmentation (Appendix F).

Inference. The same canonicalization is applied to every empirical network before inference, in the zero-shot benchmark (Appendix H), the single-instance reconstructions (Appendix I), and the

EEG analysis. Dataset-native orderings, whether anatomical, temporal or spatial, are never used; all reported numbers are produced under the metadata-free ordering. The MMD statistics of the benchmark are permutation-invariant, so canonicalizing the model input leaves the reference side of every comparison unchanged.

F Training Details

F.1 Data Generation

We generate synthetic training pairs $(A, \{p^{(1)}, l^{(1)}\})$ by constructing K-level parameters $l^{(K)}$ and $p^{(K)}$ via Kronecker products of randomly sampled base measures. The base parameters are sampled as follows. For the community distribution $l^{(1)} \in \Delta^{M-1}$, each entry is initialized to a minimum value of 0.1 to ensure that every community has non-negligible measure, and the remaining probability mass $(1-0.1M)$ is distributed by drawing M uniform random values and normalizing them to sum to the residual. The interaction matrix $p^{(1)} \in (0,1)^{M\times M}$ is sampled entry-wise from $\mathrm{Uniform}(0.1, 1.0)$, ensuring strict positivity as required by the identifiability result (Proposition 1). Parameter sets with $|l_0^{(1)} - l_1^{(1)}| \leq 0.01$ are rejected. The targets are ordered by expected degree, so community 0 is the higher-degree community.

Nodes are assigned to communities from a multinomial distribution parameterized by $l^{(K)}$, and edge weights are drawn as $e_{ij} \sim \mathrm{Geometric}(1 - p_{qh}^{(K)})$. Non-zero edge weights are log-binned to form the input adjacency matrix:

$$A_{ij} = \lfloor \log_2(e_{ij}) \rfloor + 1 \quad \text{for} \quad e_{ij} > 0 \tag{S21}$$

Inputs are canonicalised as in Appendix E.1 before log-binning.

Augmentation. Each training graph is augmented with probability 0.5 by node dropout, edge dropout and spurious edges, each at a rate drawn from $\mathrm{Uniform}(0, 0.3)$. The edge masks are symmetric with zero diagonal. Augmented graphs are binarized and are not re-sorted after augmentation.

F.2 Loss Function

The model is trained with a composite loss. The interaction matrix is optimized via binary cross-entropy with logits ($\mathcal{L}_{\text{BCE}}$) on the raw MLP outputs $\hat{Z}_p$, while the node distribution $\hat{l}^{(1)} \in \Delta^{M-1}$ is optimized via KL divergence ($\mathcal{L}_{\text{KL}}$):

$$\mathcal{L}(\theta^{(1)}, \hat{\theta}^{(1)}) = \mathcal{L}_{\text{BCE}}(p^{(1)}, \hat{Z}_p) + \mathcal{L}_{\text{KL}}(l^{(1)} \parallel \hat{l}^{(1)}) \tag{S22}$$

F.3 Optimization and Hyperparameters

The model is trained for 2000 epochs using AdamW with a base learning rate of 10^{-3}, beginning with a 10-epoch linear warmup from 10^{-5} to 10^{-3}, followed by cosine annealing to 10^{-5} over the remaining epochs. The batch size is 64 (32 per GPU on two GPUs).

Each epoch draws 2000 new graphs from the generator. Training graphs have $N \in \{128, 512, 1024\}$ and $K \in \{2, 4, 6\}$. Validation uses 100 fixed graphs at $K = 4$ and $N = 512$, and we keep the epoch with the lowest validation MAE of $p^{(1)}$ plus that of $l^{(1)}$ (epoch 1816). The MWNOT architecture uses the following hyperparameters: query/key dimension $D = 128$, latent feature/value dimension $S = 128$, $L = 4$ wavelet decomposition scales, and wavelet order $k = 6$ (corresponding to order-6 Legendre polynomial bases). The number of base communities is $M = 2$ throughout. Recovery error (Figure 4) is computed on 100 new, unaugmented graphs per (N, K) (random seed 42 for the whole run), with the depth sweep at $N = 1024$, after matching predicted to true communities by the best of the $M!$ permutations.

F.4 Compute resources.

Training was performed on two NVIDIA V100 GPUs and completed in approximately 8 hours. Inference is fast enough to run on a single GPU or modern CPU: per-graph inference time is reported

in Table 2 ("Time (ms)" column) as one forward pass on the cropped input, averaged over 10 repeats after one warm-up pass, excluding normalisation, sorting and log-binning.

G Expressivity Sweep Details

For both sweeps, we set $M = 2$, $K = 4$, and $N = 1024$, averaging over 10 realizations per setting. The heterogeneity sweep uses $\mathbf{l}^{(1)} = [0.25, 0.75]$ and interpolates $\mathbf{p}^{(1)}$ from a uniform matrix $[[0.5, 0.5], [0.5, 0.5]]$ to a heterogeneous matrix $[[0.99, 0.05], [0.05, 0.01]]$ as w_{het} varies from 0 to 1 over 30 steps. Panel (d) of Figure 3 is drawn from a separate sweep, in which $\mathbf{p}^{(1)}$ is interpolated in 12 steps towards each of three targets (left-skewed $[[0.99, 0.2], [0.2, 0.01]]$, symmetric $[[0.99, 0.01], [0.01, 0.99]]$ and right-skewed $[[0.01, 0.6], [0.6, 0.99]]$), with one realization per step and $q \in [-3.5, 3.5]$. The structure sweep uses $\mathbf{l}^{(1)} = [0.4, 0.6]$ and interpolates between a disassortative matrix $[[0.01, 0.99], [0.99, 0.01]]$ at $w_{\text{struct}} = -1$, the uniform matrix at $w_{\text{struct}} = 0$, and an assortative matrix $[[0.99, 0.01], [0.01, 0.99]]$ at $w_{\text{struct}} = +1$ over 21 steps. In panels (b) and (c), degree assortativity is computed on the directed weighted graph, and Louvain modularity and the normalised-Laplacian spectral gap on the weighted undirected graph, with isolated nodes removed; $\mathbf{p}^{(1)}$ is clamped to $[0.01, 0.99]$. The multifractal spectra in these sweeps use q grids of 51 points on $[-10, 10]$ or 40 points on $[-8, 8]$, instead of the grid of Appendix J. Panel (a)'s degree-distribution regimes use distinct hand-tuned $(\mathbf{l}^{(1)}, \mathbf{p}^{(1)}, K)$ configurations to target each canonical shape (full parameters in the released code).

The Lipschitz–Hölder exponent α measures local singularity strength: small α corresponds to hubs, large α to peripheral nodes. $f(\alpha)$ gives the fractal dimension of the set of nodes at each α. The width $\Delta\alpha = \alpha_{\max} - \alpha_{\min}$ summarizes multiscale heterogeneity. Full definitions are in Appendix J.

H Zero-shot benchmark: evaluation protocol and full results

This appendix states the protocol behind Table 1 and reports the full table with seed-to-seed variability (Table S1). We follow the evaluation protocol of the LGGM benchmark (Wang et al., 2025a), such as the test splits, metrics, and released evaluation code. The paragraphs below state how our model is applied within that protocol.

Data. We use the twelve domains and the held-out test graphs of the LGGM benchmark (Wang et al., 2025a) processed exactly as the benchmark's evaluation pipeline (undirected, self-loops removed). For ECO, as in the released evaluation code, the reference combines the training, validation and test splits (251 graphs, six unique).

Inference. The inverse operator receives the canonicalised (Appendix E.1) binary adjacency matrix, truncated to the largest dyadic size $2^{\lfloor \log_2 N \rfloor}$ (graphs with fewer than $2^L = 16$ nodes are zero-padded to 16 nodes) as required by the multiwavelet decomposition. One forward pass returns $\hat{\theta}^{(1)} = (\hat{l}^{(1)}, \hat{p}^{(1)})$. No per-graph optimisation is performed and $\hat{\theta}^{(1)}$ is used as returned.

Recursion depth. The depth K used to realise a graph from $\hat{\theta}^{(1)}$ is set per reference graph by the closed-form density relation of Appendix B, $\rho_{\text{emp}} = \rho(W_1)^K$ with $\rho(W_1) = \sum_{ij} \hat{l}_i \hat{l}_j \hat{p}_{ij}$, solved as $K^\star = \text{round}\big(\log \rho_{\text{emp}} / \log \rho(W_1)\big)$ and clamped to $[2, 6]$. Here ρ_{emp} is the number of non-zero entries divided by N^2, diagonal included, on the full uncropped network; rounding is half-to-even, and $K = 4$ is used if ρ_{emp} or $\rho(W_1)$ lies outside $(0, 1)$. $K^\star$ depends only on $\hat{\theta}^{(1)}$ and the observed edge density. It is fixed before any graph is generated. Therefore, no evaluation metric is minimised over K.

Generation and metrics. For each reference graph and each seed, one graph of the same size is sampled from $\hat{\theta}^{(1)}$ at depth $K^\star$. These graphs are directed and may contain self-loops. Following the benchmark's protocol, they are binarized, symmetrised by union ($A + A^\top > 0$) and stripped of self-loops before calculating the reconstruction metrics. We repeat the sampling for ten seeds (42–51) and report mean and standard deviation. DiGress and LGGM-X entries are the values published with

the benchmark. Ours-Mono is the ablation of Ours, trained with $p^{(1)}$ fixed at 0.5 and evaluated under the identical protocol.

I SINGLE-INSTANCE RECONSTRUCTION: PROTOCOL

This appendix states the protocol behind Table 2. Its metrics are defined in Appendix J. Each dataset is a single observed network. The comparison asks the reconstruction fidelity of networks generated from the zero-shot estimate $\hat{\theta}^{(1)}$, with a per-instance expectation–maximisation fit (WMGM) as a reference for what direct optimisation on the same network achieves.

Data. We use six networks, each observed once. Four are directed (C. elegans, Drosophila medulla, Allen mouse and P2P-Gnutella). Two are undirected (Power Grid and Minnesota Roads). Only C. elegans and Allen mouse carry edge weights. The C. elegans file is a 383×575 connection table whose rows are the 383 neurons. We keep only the neuron-to-neuron connections, the leading 383×383 block. The Allen mouse, Drosophila and C. elegans networks contain self-loops (163, 105 and 8), which are kept. They enter the edge density, the node strengths used by the multifractal MSE and the degrees, where each self-loop counts once. Before use, we divide every weight by the smallest non-zero weight in that network, so the weakest edge has weight 1. We then order the nodes as described in Appendix E.1. This full-size network is the reference that every generated network is compared against. Both methods receive this network after log-binning (Eq. (S21)). WMGM is fitted on the full network. Our operator receives the crop to the largest dyadic size for compatibility with multiwavelet transform, as described in Appendix E.1.

Recursion depth. As in Appendix H, $K^\star$ is set by the closed-form density relation from $\hat{\theta}^{(1)}$ and the observed edge density, and is fixed before generation. For WMGM, which is fitted separately at each depth $K \in \{2, \dots, 6\}$, the reported depth is the one at which the fitted parameters are self-consistent with the same relation, i.e. the K minimising $|K^\star(\hat{\theta}^{(1)}_K) - K|$. No evaluation metric enters the choice for either method.

Generation and evaluation. For each of ten seeds (42–51), one weighted network of the original size is sampled from each method's parameters at its depth. Samples are directed and may contain self-loops. The degree and spectral distances are computed on the binarized sample with self-loops kept. The sample is symmetrised by union for the spectral distance and, for the undirected references (Power Grid and Minnesota Roads), also for the degree distance. The multifractal MSE uses the out-strengths of the directed weighted sample. Each sample is compared with the reference network using multifractal MSE, normalised degree Wasserstein distance, and spectral Wasserstein distance of the normalised Laplacian. We report mean and standard deviation over seeds.

WMGM fitting. WMGM is run on the same canonicalised, log-binned matrix at full size with $M = 2$, up to 1000 EM iterations and an Adam step size of 10^{-3}, once per depth and seed. Fitting stops early when the change in the evidence lower bound falls below 10^{-4}, and the step size is halved when the objective plateaus (patience 50, cooldown 20, minimum 10^{-6}). The parameters are initialized as follows: $l^{(1)}$ as uniform, $p^{(1)}$ from $\mathrm{Uniform}(0, 1)$, and the community responsibilities from $\mathrm{Dirichlet}(1)$. $p^{(1)}$ is clamped to $[10^{-6}, 1 - 10^{-6}]$. Ties in the depth rule are resolved to the smallest K. Its reported time is the mean wall-clock time per fit over the five depths and ten seeds, measured on the same GPU as our inference.

J DEFINITIONS OF EVALUATION METRICS FOR SINGLE OBSERVATION DATASETS

J.1 MULTIFRACTAL SPECTRUM ($f(\alpha)$) ESTIMATION

To estimate the continuous multifractal spectrum of the networks, we apply the Chhabra-Jensen method. Let A be the adjacency matrix of a network with N active nodes, and let $d_i = \sum_j A_{ij}$ be the degree of node i. We first define the degree probability distribution $P_i = d_i / \sum d_i$.

Table S1: Full version of Table 1 with seed-to-seed standard deviations. Lower is better.

Domain	Method	DEG	CC	Spec	Orb
FB	DiGress	0.3376	0.6298	0.0797	0.3593
	LGGM-X	0.4723	0.6843	0.2924	0.7555
	Ours-Mono	0.2782 ± 0.0017	0.3045 ± 0.0166	0.0308 ± 0.0004	0.0412 ± 0.0008
	Ours	**0.0589 ± 0.0027**	**0.0500 ± 0.0013**	**0.0284 ± 0.0008**	0.0403 ± 0.0007
ASN	DiGress	0.1496	0.3258	0.1506	0.4420
	LGGM-X	**0.0281**	0.2440	**0.0830**	**0.0618**
	Ours-Mono	0.1350 ± 0.0070	0.1375 ± 0.0053	0.1484 ± 0.0056	0.5197 ± 0.0411
	Ours	0.0780 ± 0.0081	0.1281 ± 0.0096	0.1003 ± 0.0049	0.3012 ± 0.0254
EMAIL	DiGress	0.2192	0.6012	0.0702	0.3416
	LGGM-X	0.0751	0.2364	0.0768	0.3089
	Ours-Mono	0.1396 ± 0.0017	0.0921 ± 0.0041	0.0600 ± 0.0011	0.0765 ± 0.0023
	Ours	**0.0631 ± 0.0037**	**0.0529 ± 0.0005**	**0.0563 ± 0.0021**	**0.0725 ± 0.0002**
WEB	DiGress	0.2556	0.6186	0.1877	0.6045
	LGGM-X	**0.0648**	0.3961	**0.0549**	**0.1127**
	Ours-Mono	0.1553 ± 0.0033	0.1456 ± 0.0038	0.1749 ± 0.0009	0.3319 ± 0.0096
	Ours	0.0692 ± 0.0024	**0.0795 ± 0.0021**	0.1145 ± 0.0023	0.2760 ± 0.0076
ROAD	DiGress	0.3705	0.8226	0.2801	0.7198
	LGGM-X	**0.0713**	0.2193	**0.0987**	**0.2986**
	Ours-Mono	0.4300 ± 0.0025	0.6884 ± 0.0091	0.3142 ± 0.0016	0.7433 ± 0.0076
	Ours	0.1107 ± 0.0039	**0.1850 ± 0.0140**	0.1446 ± 0.0047	0.6027 ± 0.0194
POWER	DiGress	0.3726	0.4582	0.3270	1.4732
	LGGM-X	**0.0119**	**0.1293**	**0.0373**	**0.0754**
	Ours-Mono	0.1963 ± 0.0037	0.3947 ± 0.0184	0.2809 ± 0.0036	0.9536 ± 0.0265
	Ours	0.1660 ± 0.0063	0.2992 ± 0.0142	0.2265 ± 0.0066	0.6956 ± 0.0293
BIO	DiGress	0.2712	0.5202	0.1127	0.3188
	LGGM-X	0.1081	0.2696	0.0900	0.2053
	Ours-Mono	0.0879 ± 0.0013	0.1467 ± 0.0078	0.0827 ± 0.0012	0.1584 ± 0.0077
	Ours	**0.0442 ± 0.0016**	**0.0605 ± 0.0033**	**0.0629 ± 0.0021**	**0.1090 ± 0.0053**
ECON	DiGress	0.2987	0.4841	0.2162	0.3834
	LGGM-X	0.1213	0.0920	0.1120	0.1086
	Ours-Mono	0.1011 ± 0.0011	0.1092 ± 0.0066	0.0707 ± 0.0010	0.0916 ± 0.0028
	Ours	**0.0780 ± 0.0018**	**0.0474 ± 0.0030**	**0.0199 ± 0.0007**	**0.0598 ± 0.0026**
RT	DiGress	0.4164	**0.1327**	0.4147	0.5957
	LGGM-X	**0.0525**	0.1429	**0.1330**	0.2219
	Ours-Mono	0.2911 ± 0.0040	0.4537 ± 0.0385	0.3705 ± 0.0039	0.3612 ± 0.0247
	Ours	0.1999 ± 0.0081	0.4070 ± 0.0341	0.2345 ± 0.0120	0.2452 ± 0.0280
COL	DiGress	0.2473	0.5826	0.2314	0.7679
	LGGM-X	**0.0736**	0.5769	**0.0895**	**0.0988**
	Ours-Mono	0.2143 ± 0.0035	0.3022 ± 0.0100	0.2329 ± 0.0020	0.7161 ± 0.0080
	Ours	0.1020 ± 0.0021	**0.1683 ± 0.0062**	0.1863 ± 0.0034	0.6032 ± 0.0156
ECO	DiGress	0.5431	0.7915	0.2338	0.6045
	LGGM-X	0.4753	0.3904	0.3194	0.3934
	Ours-Mono	0.1744 ± 0.0010	0.3025 ± 0.0038	0.0436 ± 0.0005	0.2044 ± 0.0001
	Ours	**0.0632 ± 0.0026**	**0.2298 ± 0.0003**	**0.0326 ± 0.0004**	0.2040 ± 0.0000
CITATION	DiGress	0.2527	0.7790	0.1315	0.4966
	LGGM-X	0.1348	0.7257	0.1160	0.4981
	Ours-Mono	0.1364 ± 0.0013	0.2654 ± 0.0093	0.0917 ± 0.0007	0.1601 ± 0.0052
	Ours	**0.0407 ± 0.0013**	**0.1150 ± 0.0019**	**0.0621 ± 0.0017**	**0.1010 ± 0.0026**
ALL	DiGress	0.3112	0.5622	0.2030	0.5923
	LGGM-X	0.1408	0.3422	0.1253	**0.2616**
	Ours-Mono	0.1950 ± 0.0027	0.2785 ± 0.0111	0.1584 ± 0.0019	0.3632 ± 0.0114
	Ours	**0.0895 ± 0.0037**	**0.1519 ± 0.0075**	**0.1057 ± 0.0035**	0.2759 ± 0.0114

DEG, CC, Spec, Orb: MMD of degree, clustering coefficient, Laplacian eigenvalues and orbit counts. DiGress and LGGM-X are published values. Ours and Ours-Mono are mean ± SD over 10 seeds; the SD of the ALL row is the mean of the twelve per-domain SDs. Bold marks the best method where the mean ± SD intervals do not overlap. On ECO the orbit-count kernel saturates, so the orbit MMD is a constant set by the reference graphs for any distinct, dense generated graphs; it is not used to mark the best method.

Table S2: Full version of Table 2 with seed-to-seed standard deviations.

Domain	Method	Time (ms)	MFrac	DegW	SpecW
C. elegans	WMGM	2060 ± 436	2.68e-2 ± 2.17e-2	**5.67e-4 ± 2.55e-4**	**6.97e-2 ± 3.47e-3**
	Ours	**3.76 ± 0.97**	**2.19e-4 ± 4.02e-5**	1.20e-3 ± 2.08e-5	1.27e-1 ± 6.92e-4
Drosophila	WMGM	1930 ± 77	7.35e-3 ± 8.56e-3	2.29e-4 ± 1.18e-4	**6.41e-2 ± 1.62e-2**
	Ours	**14.1 ± 0.03**	3.84e-3 ± 3.68e-4	2.69e-4 ± 6.71e-6	1.18e-1 ± 1.42e-3
Allen Mouse	WMGM	1050 ± 243	**1.52e-4 ± 1.51e-5**	1.38e-4 ± 3.76e-5	3.99e-3 ± 9.20e-3
	Ours	**2.95 ± 0.04**	7.86e-3 ± 2.00e-3	9.86e-5 ± 8.96e-6	6.82e-3 ± 8.37e-4
P2P Gnutella	WMGM	2540 ± 131	5.19e-3 ± 3.07e-3	1.41e-4 ± 2.78e-5	**8.09e-2 ± 2.23e-2**
	Ours	**198 ± 0.22**	3.56e-3 ± 3.43e-4	1.23e-4 ± 8.12e-7	1.24e-1 ± 9.60e-4
Power Grid	WMGM	1810 ± 201	5.88e-4 ± 9.86e-4	4.23e-5 ± 1.45e-5	**1.81e-1 ± 6.52e-2**
	Ours	**198 ± 0.05**	1.24e-3 ± 8.78e-5	**2.54e-5 ± 4.88e-7**	2.58e-1 ± 2.25e-3
MN Roads	WMGM	1490 ± 280	**1.73e-3 ± 2.64e-4**	**5.92e-5 ± 6.16e-6**	1.86e-1 ± 4.97e-2
	Ours	**50.9 ± 1.59**	2.42e-3 ± 2.00e-5	1.24e-4 ± 1.91e-6	1.84e-1 ± 5.11e-3

MFrac: multifractal MSE; DegW: degree Wasserstein; SpecW: spectral Wasserstein. Values are mean ± SD over 10 seeds. Bold marks the lower value where the two methods' mean ± SD intervals do not overlap.

To isolate different scaling regimes, we define a one-parameter family of normalized measures, $\mu_i(q)$, parameterized by the moment $q \in \mathbb{R}$:

$$\mu_i(q) = \frac{P_i^q}{\sum_{j=1}^{N} P_j^q} \tag{S23}$$

By varying q, we heavily weight the massive hubs ($q > 0$) or the low-degree periphery ($q < 0$). The Lipschitz-Hölder exponent $\alpha(q)$ and its corresponding fractal dimension $f(\alpha(q))$ are then computed parametrically as:

$$\alpha(q) = -\frac{1}{\log N} \sum_{i=1}^{N} \mu_i(q) \log P_i \tag{S24}$$

$$f(\alpha(q)) = -\frac{1}{\log N} \sum_{i=1}^{N} \mu_i(q) \log \mu_i(q) \tag{S25}$$

The **Multifractal MSE** is calculated as the mean squared error between the empirical and synthetic continuous curves, $f_{emp}(\alpha)$ and $f_{syn}(\alpha)$, interpolated over a shared α support space. We use 30 values of q evenly spaced in $[-4, 4]$ and interpolate both curves linearly on 50 points spanning the overlap of their α ranges.

J.2 Degree Wasserstein Distance

To evaluate the absolute local hub structure, we utilize the 1-Wasserstein distance (Earth Mover's Distance) between the sets of empirical and synthetic degrees. Let $U(x)$ and $V(x)$ be the cumulative distribution functions of the empirical degrees d^{emp} and synthetic degrees d^{syn}, respectively. The unscaled degree distance is defined as:

$$W_1(d^{emp}, d^{syn}) = \int_{-\infty}^{\infty} |U(x) - V(x)| \, dx \tag{S26}$$

For cross-dataset comparisons independent of absolute scale (as presented in the main text), we compute the Normalized Degree Wasserstein by transforming the degrees into probability mass functions: $\tilde{d}_i = d_i / \sum d_j$ prior to calculating the Wasserstein integral.

J.3 Spectral Wasserstein Distance

To evaluate the global community structure, bottlenecking, and information flow of the networks, we analyze the eigenspectrum of the Normalized Laplacian matrix. The Normalized Laplacian is defined as:

$$\mathcal{L} = I - D^{-1/2} A D^{-1/2} \tag{S27}$$

where I is the identity matrix and D is the diagonal degree matrix. Because the eigenvalues of $\mathcal{L}$ are strictly bounded within the interval $[0, 2]$ regardless of network size or absolute density, they provide a perfectly scaled invariant measure of global topology.

Let λ^{emp} and λ^{syn} denote the sorted sets of eigenvalues derived from the empirical and synthetic Normalized Laplacians. The Spectral Wasserstein distance is calculated as the 1-Wasserstein distance between these two discrete distributions:

$$W_1(\lambda^{emp}, \lambda^{syn}) = \inf_{\gamma \in \Gamma(\lambda^{emp}, \lambda^{syn})} \int \int |x - y| \, d\gamma(x, y) \tag{S28}$$

where Γ denotes the collection of all joint probability measures with marginals equivalent to the empirical and synthetic eigenvalue distributions.

The degree and spectral distances are computed on binarized graphs. For directed networks, the degree distance uses out-degrees and the spectral distance uses the union-symmetrised graph. Standard deviations over seeds (Tables S1 and S2) are population SDs; Table 3 reports sample SDs across subjects.

K EEG Analysis

Data. We use the propofol sedation dataset of Chennu et al. (2016), which includes 20 healthy volunteers who were recorded at rest with 91-channel scalp EEG (250 Hz) across four sessions: before propofol administration (baseline), under mild sedation, under moderate sedation, and after recovery.

Network construction. Each epoch is band-pass filtered to the alpha band (8–15 Hz, fourth-order Butterworth, zero-phase) and Hilbert-transformed. Functional connectivity between every pair of channels is the weighted phase-lag index (wPLI) computed per epoch and averaged over the epochs of the session (Vinck et al., 2011). Edges are retained by a surrogate test. For each epoch we generate 1000 phase-randomised surrogates of every channel (Fourier amplitudes preserved, phases drawn uniformly), compute their wPLI matrices, and average them over epochs. An edge is kept, with its wPLI value as weight, if the observed session-mean wPLI exceeds the 95th percentile of the surrogate distribution for that channel pair.

Inference. Each network is processed as in Appendix I. Weights are normalised so that the weakest edge has weight 1, nodes are canonicalised (Appendix E.1), the matrix is cropped to 64 nodes, which removes the 27 lowest-strength channels, and log-binned. One forward pass of the operator returns $\hat{\theta}^{(1)}$. From $\hat{\theta}^{(1)} = (\hat{l}^{(1)}, \hat{p}^{(1)})$ we report two scalar summaries. One is the assortativity index $\hat{p}_{00} + \hat{p}_{11} - \hat{p}_{01} - \hat{p}_{10}$, which is positive when within-class connections dominate. The other is multifractal width $\Delta\alpha = \log_2(\hat{p}_{\max}/\hat{p}_{\min})$, which is the width of the spectrum of Appendix B for $M = 2$. Classical metrics (mean clustering, mean shortest path, greedy modularity, spectral gap of the normalised Laplacian) are computed on the same networks before cropping and after binarisation ($A_{ij} > 0$). The shortest path and spectral gap use the largest connected component.

Group comparison. The four sessions are compared as groups of 20 networks. For each measure we report the following: 1) mean and SD per state 2) Cohen's d with pooled SD for each state minus moderate sedation and for baseline minus recovery 3) a two-sided Mann–Whitney U test per contrast with Holm correction over the four contrasts.

Predictive validity. To test whether the inferred rule summaries generalize to new individuals, we train a logistic regression on the two $\hat{\theta}^{(1)}$-derived summaries to distinguish moderate sedation from each other state, holding out one subject at a time. Features are standardised within each training fold, and the model uses the default L2 penalty ($C = 1$). Predictions are pooled across held-out subjects and scored by AUC with 95% bootstrap intervals over the 20 subjects (1000 resamples). The AUC is 0.69 [0.55, 0.83] for moderate sedation versus baseline, 0.72 [0.58, 0.85] versus mild sedation and 0.75 [0.62, 0.89] versus recovery. All three intervals exclude chance (0.5).

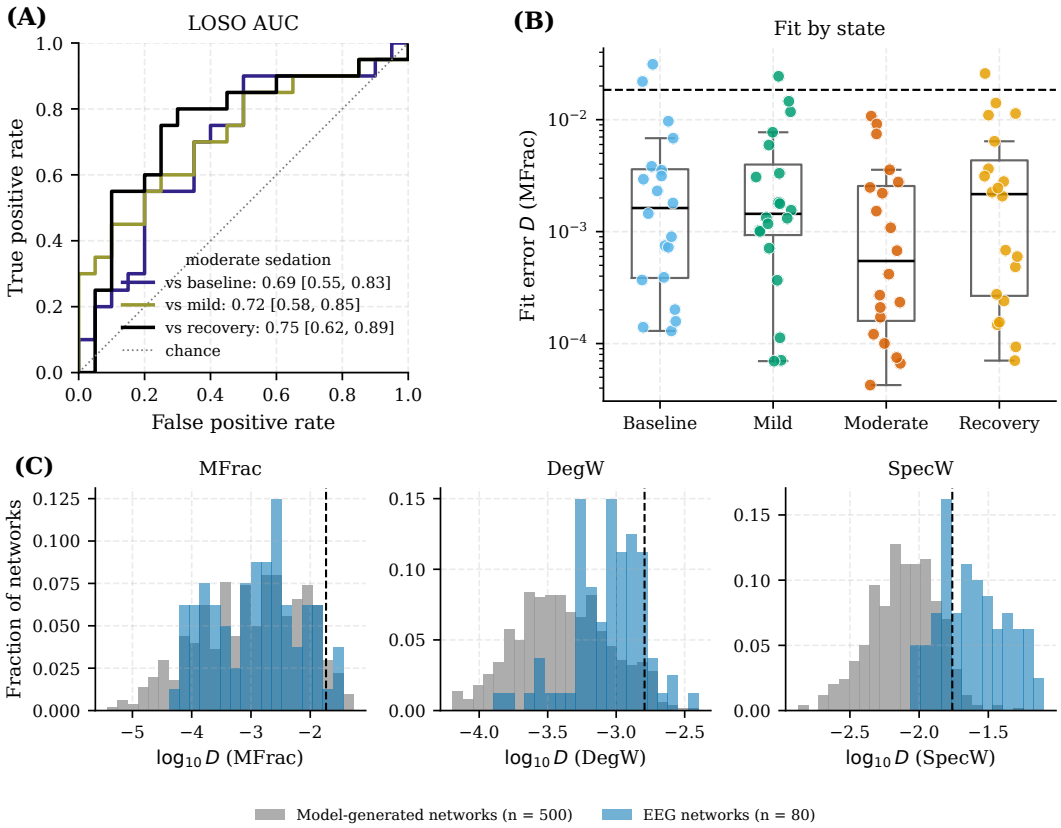

Figure S2: **Supplementary EEG analyses. (A)** Predictive validity. ROC curves of the leave-one-subject-out (LOSO) logistic regression. The legend gives the AUC with its 95% bootstrap interval over subjects. Dotted line represents chance. **(B)** Fit by state. Reconstruction error D (multifractal MSE, mean over ten generated networks) of every EEG network, grouped by sedation state ($n = 20$ per state). Boxes represent median and interquartile range. Whiskers represent $1.5\times$IQR. Dashed line represents the pass threshold of (C) $D = 0.0185$. (Kruskal–Wallis $p = 0.40$; Friedman over subjects $p = 0.42$; two-sided Mann–Whitney tests of moderate sedation against each other state, Holm-corrected, $p = 0.65$ for all three; baseline vs recovery $p = 0.86$). **(C)** Synthetic reference range. Distribution of $\log_{10} D$ for 500 model-generated networks whose density lies in the EEG range (gray) and for the 80 EEG networks (blue), for the multifractal MSE (MFrac; the decision metric), the normalised degree Wasserstein distance (DegW) and the spectral Wasserstein distance (SpecW). Dashed lines represent the 95th percentile of the model-generated networks (0.0185, 0.0016 and 0.017). The number of EEG networks below the line pass are as follows: 76 of 80 on MFrac, 71 of 80 on DegW, and 27 of 80 on SpecW.

Synthetic reference range. We ask whether the reconstruction error of each EEG network is within the range seen when the model is true. We draw parameter sets from the training prior and a depth K uniformly from $\{2, \ldots, 6\}$, and generate one 91-node network from each. Unlike the EEG networks, which are symmetric, loop-free and wPLI-weighted, these networks are directed, may contain self-loops and have integer weights. As in Table 2, the degree and spectral distances are computed on binarized, union-symmetrised graphs with self-loops kept, and the multifractal MSE on the directed weighted samples. We keep a network only if its edge density lies in the observed EEG range, until 500 networks are kept. Each network, synthetic or EEG, is then processed as described above, given to MWNOT as inputs, and ten networks are generated from each inferred rule (seeds 42 to 51). The generated graphs are each compared with the input network using the multifractal MSE of Appendix J. The mean over the ten comparisons is the error D. The pass threshold is the 95th percentile of D over the 500 synthetic networks, 0.0185. The model's fit on a given EEG is considered acceptable if its D is below the threshold.

Of the 80 EEG networks, 76 pass (18 baseline, 19 mild, 20 moderate and 19 recovery). The median D of the EEG networks, 0.00139, is close to the synthetic median, 0.00130. The degree Wasserstein and spectral Wasserstein distances are reported for completeness but do not enter the decision. Under the same rule, 71 networks pass on the degree distance and 27 on the spectral distance. The spectral gap between real and synthetic networks is consistent with Table 2, where spectral error is also the weakest term.

Fit and sedation state. State differences in the inferred rule could reflect worse fit under moderate sedation rather than a real change. To rule this out, we compared fit across states. We use D as the fit score. Its median is 0.0016 at baseline, 0.0014 under mild sedation, 0.0005 under moderate sedation and 0.0022 at recovery. A Kruskal–Wallis test across the four states gives $p = 0.40$. A Friedman test over the 20 subjects gives $p = 0.42$. Two-sided Mann–Whitney tests of moderate sedation against each other state, Holm-corrected over the four contrasts of Table 3, give $p = 0.65$ for all three. Baseline versus recovery gives $p = 0.86$. We found no evidence that fit differs across states.

L Compression of a periodic multirule generator

Proposition 2 (Compression of a periodic multirule generator)**.** *For each scale k, let*

$$l^{[k]} \in \mathrm{int}(\Delta^{M_k-1}), \qquad p^{[k]} \in (0,1)^{M_k \times M_k},$$

and define the multirule recursive expansion

$$l^{(K)} = \bigotimes_{k=1}^{K} l^{[k]}, \qquad p^{(K)} = \bigotimes_{k=1}^{K} p^{[k]}.$$

Assume that the rule sequence is periodic with period R, so that

$$(l^{[k+R]}, p^{[k+R]}) = (l^{[k]}, p^{[k]}) \qquad \textit{for all } k,$$

and let $K = mR$. Define

$$\tilde{l}^{(1)} := \bigotimes_{r=1}^{R} l^{[r]}, \qquad \tilde{p}^{(1)} := \bigotimes_{r=1}^{R} p^{[r]}, \qquad \tilde{M} := \prod_{r=1}^{R} M_r.$$

Then

$$\tilde{l}^{(1)} \in \mathrm{int}(\Delta^{\tilde{M}-1}), \qquad \tilde{p}^{(1)} \in (0,1)^{\tilde{M}\times\tilde{M}},$$

and the original multirule generator is exactly representable as

$$l^{(K)} = \underbrace{\tilde{l}^{(1)} \otimes \cdots \otimes \tilde{l}^{(1)}}_{m \text{ times}}, \qquad p^{(K)} = \underbrace{\tilde{p}^{(1)} \otimes \cdots \otimes \tilde{p}^{(1)}}_{m \text{ times}}.$$

Hence a periodic multirule recursive generator reduces exactly to a single-basis generator with effective branching number

$$\tilde{M} = \prod_{r=1}^{R} M_r.$$

Proof. Since each $l^{[r]}$ is a strictly positive probability vector, their Kronecker product $\tilde{l}^{(1)}$ is again a strictly positive probability vector, now of dimension $\prod_{r=1}^{R} M_r$. Likewise, $\tilde{p}^{(1)}$ has dimension $\tilde{M} \times \tilde{M}$ and entries in $(0,1)$.

Because the rule sequence is periodic and $K = mR$, the full expansion can be grouped into m identical blocks:

$$l^{(K)} = \Big(\bigotimes_{r=1}^{R} l^{[r]}\Big) \otimes \cdots \otimes \Big(\bigotimes_{r=1}^{R} l^{[r]}\Big) = (\tilde{l}^{(1)})^{\otimes m}.$$

The same argument gives

$$p^{(K)} = (\tilde{p}^{(1)})^{\otimes m}.$$

Therefore the multirule generator is exactly representable as an m-level recursion of the single effective basis $(\tilde{l}^{(1)}, \tilde{p}^{(1)})$. □

Obstruction to finite single-basis reduction: If a multirule recursive generator does not admit a decomposition into repeated identical finite blocks, then it cannot be represented exactly by a finite-dimensional single basis repeated across scales.

Proof. Any finite-dimensional single basis repeated across scales induces a rule sequence obtained by repetition of one finite block. The claim follows by contrapositive. □

Theorem 1 (Minimal effective branching number)**.** *Consider a multirule recursive generator of depth K,*

$$l^{(K)} = \bigotimes_{k=1}^{K} l^{[k]}, \qquad p^{(K)} = \bigotimes_{k=1}^{K} p^{[k]},$$

where

$$l^{[k]} \in \operatorname{int}(\Delta^{M_k-1}), \qquad p^{[k]} \in (0,1)^{M_k \times M_k}.$$

Assume that the rule sequence admits an exact single-basis reduction. Let q_* be the smallest positive divisor of K such that the sequence

$$\big((l^{[1]}, p^{[1]}), \ldots, (l^{[K]}, p^{[K]})\big)$$

can be partitioned into K/q_* consecutive identical blocks of length q_*, up to one consistent permutation of community labels.

Then the smallest effective branching number among all exact single-basis reductions is

$$\tilde{M}_{\min} = \prod_{k=1}^{q_*} M_k.$$

Moreover, the corresponding minimal effective basis is

$$\tilde{l}^{(1)}_{\min} = \bigotimes_{k=1}^{q_*} l^{[k]}, \qquad \tilde{p}^{(1)}_{\min} = \bigotimes_{k=1}^{q_*} p^{[k]},$$

up to permutation of the $\tilde{M}_{\min}$ effective labels, and satisfies

$$l^{(K)} = \big(\tilde{l}^{(1)}_{\min}\big)^{\otimes K/q_*}, \qquad p^{(K)} = \big(\tilde{p}^{(1)}_{\min}\big)^{\otimes K/q_*}.$$

Proof. Let q_* be the shortest repeating block length. Grouping each block into a single effective refinement step gives

$$\tilde{l}^{(1)}_{\min} = \bigotimes_{k=1}^{q_*} l^{[k]}, \qquad \tilde{p}^{(1)}_{\min} = \bigotimes_{k=1}^{q_*} p^{[k]},$$

which have effective dimension

$$\tilde{M}_{\min} = \prod_{k=1}^{q_*} M_k.$$

Since the sequence consists of K/q_* identical consecutive blocks,

$$l^{(K)} = \big(\tilde{l}^{(1)}_{\min}\big)^{\otimes K/q_*}, \qquad p^{(K)} = \big(\tilde{p}^{(1)}_{\min}\big)^{\otimes K/q_*}.$$

Thus an exact single-basis reduction with branching number $\tilde{M}_{\min}$ exists.

Now let

$$l^{(K)} = (\hat{l}^{(1)})^{\otimes J}, \qquad p^{(K)} = (\hat{p}^{(1)})^{\otimes J}$$

be any other exact single-basis reduction. Then the original rule sequence must decompose into J identical consecutive blocks, each of length $q = K/J$. By minimality of q_*, one has $q \geq q_*$. Therefore no exact reduction can use a shorter repeating block than q_*, and the construction above attains the smallest possible effective branching number. Hence

$$\tilde{M}_{\min} = \prod_{k=1}^{q_*} M_k.$$

□