ML Lab Research Proposal

Structural Nulls vs. Learned Generators on Sparse Connectome Graphs

Team: Ethan Wang (solo)

Date: July 22, 2026

Project: Cell2Circuit: Pareto-Front Audit of Connectome Graph Generation

Target Compute: 1x AMD Instinct MI300X GPU, approx. 2 weeks

---

1. PROBLEM STATEMENT

Learned graph generators like DiGress (Vignac et al., ICLR 2023) and DeFoG (Qin et al., ICML 2025) are increasingly applied to connectome completion and wiring-rule inference, yet no published benchmark includes the configuration model (Molloy and Reed, 1995) as a structural null. This null matches the test graph's exact degree sequence while randomizing all other structure, and has been standard in connectomics since Song et al. (2005). Our preliminary audit on MICrONS shows the configuration model Pareto-dominates every learned generator on marginal-summary axes (reachability, motifs), but the verdict flips on classifier and topological metrics. We need larger-scale experiments to confirm this finding, extend it to a second connectome (FlyWire), and test whether degree-conditioned generation can lift the Pareto front.

2. PROPOSED APPROACH

We run a Pareto-front audit comparing structural baselines (configuration model, uniform+Sinkhorn) against learned graph generators (DiGress, DeFoG, autoregressive models) on two connectomes: MICrONS Phase-1 L2/3 (505 neurons, density 0.022) and FlyWire whole-brain (139,255 neurons, density 0.001, Dorkenwald et al., Nature 2024). Evaluation uses held-out 100-node (MICrONS) and 500-node (FlyWire) subgraphs with cluster-bootstrap CIs and Benjamini-Hochberg FDR at $q = 0.05$.

The audit reveals that the bottleneck in sparse connectome generation is degree prediction, not learned edge scoring. RandomForest degree prediction cuts reachability L1 by 54% versus ridge ($p < 10^{-15}$), and Sinkhorn post-hoc degree projection destroys 92-100% of the learned model's top-E edge rankings. We propose a degree-conditioned DiGress variant that feeds predicted degree sequences directly into the generative process rather than correcting post-hoc, aiming to lift the Pareto front above the structural-null ceiling on all metric axes simultaneously.

We also replace the undirected persistent homology (gudhi, symmetrized adjacency) with directed persistent homology via flagser (Luetgehetmann et al., Algorithms 2020), which preserves edge directionality. This is methodologically necessary for a paper about directed graph generation on directed connectomes.

3. DATASET

Dataset	Source	Size	Use
MICrONS Phase-1 L2/3	Bae et al., Nature 2024	505 neurons, density 0.022	Primary audit substrate. 200 subgraphs (150 train, 50 test) at N=100.
FlyWire whole-brain	Dorkenwald et al., Nature 2024	139,255 neurons, density 0.001	Cross-substrate replication. 200 subgraphs at N=500 (adaptive cylindri

Preprocessing

- MICrONS adjacency binarized (edge if synapse count > 0). Neurotransmitter type and soma XYZ coordinates as 8-dimensional node features for degree predictors.
- FlyWire: 5-dimensional neurotransmitter vectors (GABA, GLUT, ACH, SER, non-neuronal) plus soma XYZ coordinates. Subgraph extraction via adaptive cylinder radius binary search targeting 480-520 neurons per sample.
- On-disk footprint: roughly 2 GB (adjacency + features). Well under a 1 TB scratch budget.

4. EVALUATION METRICS

Experiment	Metric	Target / Interpretation
Pareto-front audit (MICrONS + FlyWire)	3-hop reachability L1, 13-triad motif L1, PolyGraph Discrepancy, reciprocity error	Configuration model dominates learned generators on marginal-summary a
Directed persistent homology	Bottleneck + Wasserstein-1 distance (flagser H0/H1)	Determines whether undirected PH was masking a deficit or the learned-pi
Degree-conditioned generation	Same metric suite + Sinkhorn information retention (MI between learned score and output)	Degree-conditioned model matches config model on marginals AND beats
Cross-scale validation (N=50 to 500)	Dominance gap (config vs. best learned) as function of N	Identifies where learned models gain advantage at larger scales

5. COMPUTE REQUIREMENTS

Most experiments are CPU-bound (graph statistics, persistent homology). GPU compute is needed only for DiGress/DeFoG training and the degree-conditioned variant. Estimates are from measured single-run timings on existing infrastructure.

Phase	Hardware	Est. Wall-Clock	Notes
FlyWire N=500 extraction + degree prediction	CPU only	~4 hours	Subgraph extraction + RF/ridge/KNN training on 150 subsets
Directed PH (flagser + PRH)	CPU only	~6 hours	pyflagser on 50 test subsets per pipeline config. No GPU needed.
Degree-conditioned DiGress + DeFoG variant training	1x MI300X	~2 days	12 H100-hours equivalent. DiGress 30K steps + DeFoG variant.
Additional baselines (GraphRNN, GRAN, NetGAN)	1x MI300X	~1 day	8 H100-hours equivalent. Adaptation for directed graphs.
Cross-scale validation (N=50 to 500)	1x MI300X	~3 days	33 H100-hours equivalent. DiGress/SparseDiff at 5 scales.
Total	1x MI300X	~7 days within 2-week window	~53 H100-hours equivalent. Remainder for reruns.

Software Dependencies

- Python 3.10+, PyTorch 2.x with ROCm backend.
- pyflagser (or giotto-tda FlagserPersistence) for directed persistent homology.
- PyTorch Geometric for graph neural network baselines (GraphRNN, GRAN).
- No CUDA-only kernels. DiGress and DeFoG implementations use standard PyTorch operations.

6. EXPECTED DELIVERABLES

- Cross-substrate Pareto-front table: MICrONS and FlyWire results on the same metric axes, confirming (or refuting) that configuration-model dominance is substrate-general.
- Directed PH comparison: bottleneck distances under flagser vs. gudhi-symmetrized, quantifying the information loss from symmetrization.
- Degree-conditioned generation results: Sinkhorn information retention and full metric suite showing whether conditioning lifts the Pareto front above the structural-null ceiling.
- Cross-scale validation plot: dominance gap vs. subgraph size (N=50 to 500), identifying where learned models gain advantage.

• An AAAI-27 main-conference paper draft with all tables and figures generated from JSON result files.

7. RISK & MITIGATION

Risk	Likelihood	Mitigation
FlyWire N=500 subgraphs still too sparse for motif discrimination	Medium	Also extract N=200 and N=300 subsets to identify the sparsity threshold. If N=500 is insu
Degree-conditioned model collapses to configuration model (ignores learned edge scores)	Medium	Monitor mutual information between learned score and output throughout training. If MI c
GraphRNN/GRAN directed adaptation introduces bugs	Medium	Validate generated graphs have correct node count and reasonable density. Compare a
DiGress OOMs at N=500 (cross-scale experiment)	High	Use SparseDiff (Chen et al., 2023) for N >= 300. If SparseDiff unavailable, cap scale swe
Wall-clock overrun on MI300X	Low	Experiments prioritized: FlyWire + directed PH first (cheap), then degree-conditioned (cri