Geometric Inductive Biases for Diffusion-Based Graph Generation

Florian Grötschla, Saku Peltonen, Anisha Mohamed Sahabdeen, Roger Wattenhofer
ETH Zurich, Switzerland

1 Introduction

We embed graphs in a **latent Riemannian manifold** and recover edges by thresholding pairwise distances. The **latent geometry acts as the primary inductive bias**, determining which graph structures are natural to represent. By testing Euclidean, spherical, and hyperbolic spaces under a fixed architecture, we show that generation quality depends systematically on whether the curvature matches the structural properties of the target graphs.

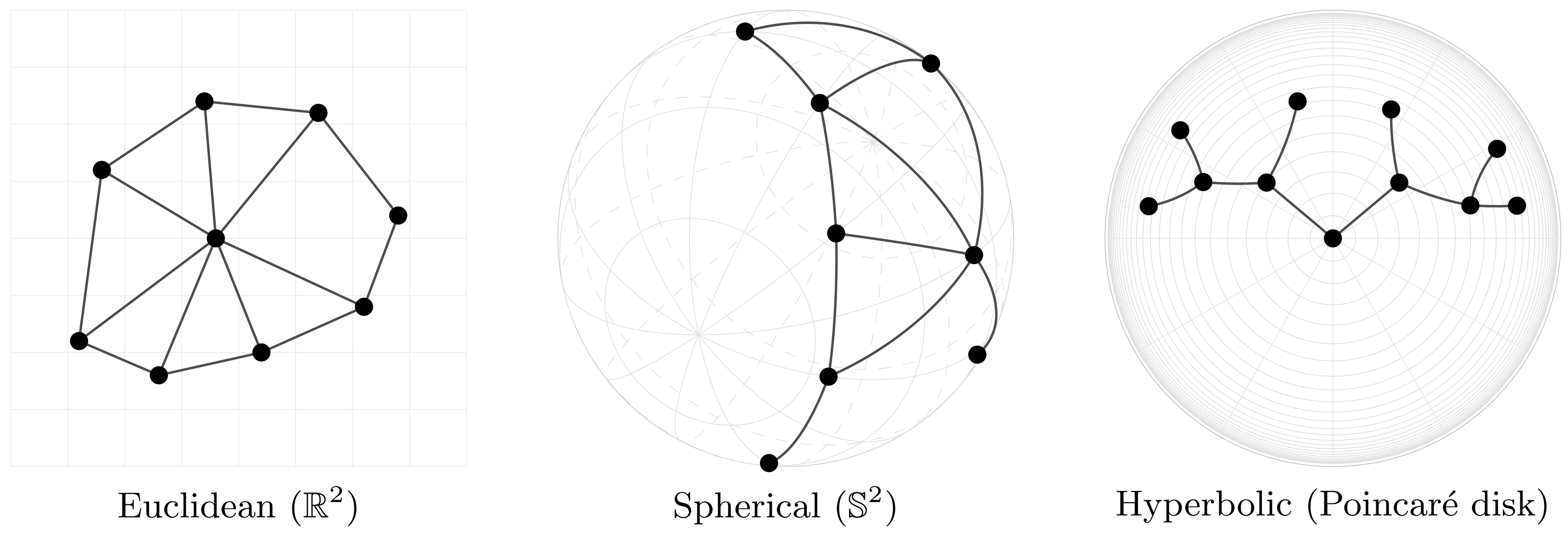

Figure 1. Inducing graphs from latent geometry. Nodes are embedded in a latent space whose metric defines the structural inductive bias. Edges are recovered by thresholding pairwise distances $d(x_i, x_j) < \tau$, where d is the metric of the chosen geometry.

2 Method Overview

- **Encode** nodes $x_i \in \mathcal{M}$ and recover edges where $d_{\mathcal{M}}(x_i, x_j) < \tau$.
- **Diffuse** with Brownian motion forward and tangent-space score matching [3] backward, fully respecting curvature.
- **Expand** using iterative local expansion [1] for variable-size graphs.

Only the **latent geometry** changes across all experiments, architecture and decoder are fixed.

3 Experimental Setup

Datasets (synthetic benchmarks from SPECTRE [2])

- **Planar**. Locally flat graphs with strong Euclidean structure.
- **SBM**. Community graphs that test separation between clusters.
- **Tree**. Hierarchical graphs, a canonical negatively curved case.
- **QM9**. Real molecular graphs.

Controlled comparison. We test Euclidean, Spherical, Hyperbolic, ℓ_1, and ℓ_∞ latent spaces while keeping the diffusion model, distance-based decoder, and local expansion mechanism fixed across all runs.

V.U.N. = validity $\times$ uniqueness $\times$ novelty. Higher is better, a perfect generator scores 100.

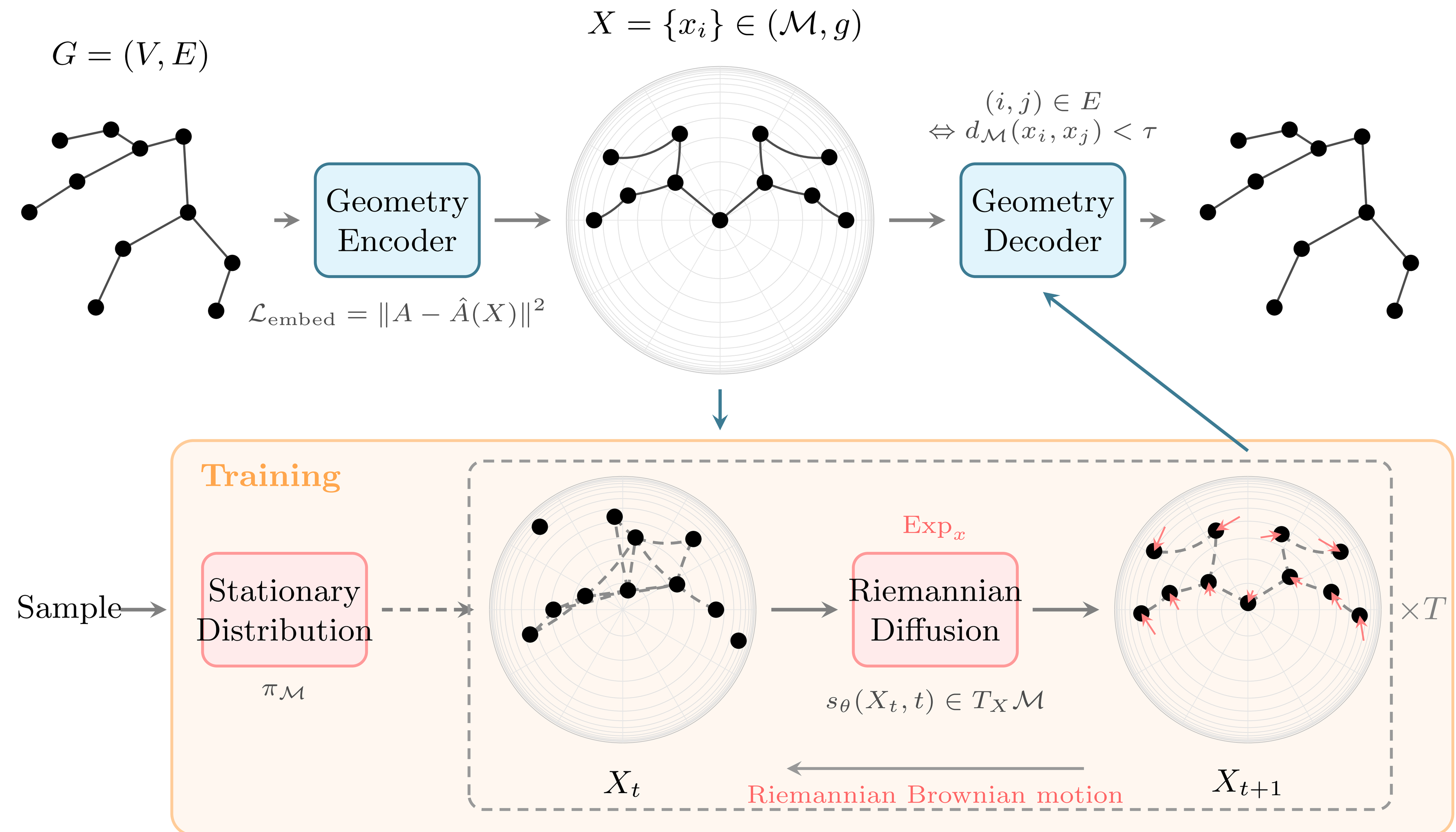

Figure 2. (Top) Encoder-decoder pipeline: a graph G is embedded into a Riemannian manifold $(\mathcal{M}, g)$, and edges are recovered by thresholding pairwise distances. **(Bottom)** Training: samples from the stationary distribution $\pi_{\mathcal{M}}$ are iteratively denoised using a learned score function s_θ, with updates via the Riemannian exponential map.

4 Results

Synthetic graph generation results. We report validity (Valid), uniqueness (Unique), novelty (Novel), and their product V.U.N. Our rows are shaded, bold marks the best result per column.

	Planar				SBM				Tree			
Model	Valid	Unique	Novel	V.U.N.	Valid	Unique	Novel	V.U.N.	Valid	Unique	Novel	V.U.N.
GraphRNN	0.0	100	100	0.0	5.0	100	100	5.0	–	–	–	–
GRAN	97.5	85.0	2.5	0.0	25.0	100	100	25.0	0.0	100	100	0.0
SPECTRE	25.0	100	100	25.0	52.5	100	100	52.5	–	–	–	–
DiGress	77.5	100	100	77.5	60.0	100	100	60.0	90.0	100	100	90.0
EDGE	0.0	100	100	0.0	0.0	100	100	0.0	0.0	7.5	100	0.0
BwR (EDP-GNN)	0.0	100	100	0.0	7.5	100	100	7.5	0.0	100	100	0.0
BiGG	62.5	85.0	42.5	5.0	10.0	100	100	10.0	100	87.5	50.0	75.0
GraphGen	7.5	100	100	7.5	5.0	100	100	5.0	95.0	100	100	95.0
LocalPGNN+ILE [1]	95.0	100	100	95.0	75.0	100	100	75.0	100	100	100	**100**
Ours (Euclidean)	97.5	100.0	100.0	**97.5**	77.5	100.0	100.0	77.5	90.8	100.0	100.0	90.8
Ours (ℓ_∞)	94.3	100.0	100.0	94.3	75.6	100.0	100.0	75.6	82.8	100.0	100.0	82.8
Ours (Spherical)	91.3	100.0	99.8	90.7	81.0	100.0	100.0	**81.0**	78.3	100.0	100.0	78.3
Ours (ℓ_1)	85.2	99.2	100.0	84.9	71.2	100.0	100.0	71.2	81.5	100.0	100.0	81.5
Ours (Hyperbolic)	84.7	100.0	100.0	84.7	66.3	100.0	100.0	66.3	100.0	99.8	100.0	**99.8**

No single geometry dominates. Matching curvature to graph structure (Euclidean for Planar, Spherical for SBM, Hyperbolic for Tree) consistently yields the best results, beating DiGress [4] and prior baselines.

5 Takeaways

- Graph generation decomposes into **geometry + diffusion + simple decoding**.
- Curvature alone drives large, predictable quality shifts under a fixed architecture.
- **Geometry is a first-class modeling choice** that controls which graph structures are easiest to generate.

References

[1] Andreas Bergmeister, Karolis Martinkus, Nathanaël Perraudin, and Roger Wattenhofer. Efficient and scalable graph generation through iterative local expansion, 2024.

[2] Karolis Martinkus, Andreas Loukas, Nathanaël Perraudin, and Roger Wattenhofer. Spectre: Spectral conditioning helps to overcome the expressivity limits of one-shot graph generators, 2022.

[3] Yang Song. Generative modeling by estimating gradients of the data distribution. `https://yang-song.net/blog/2021/score/`, 2021.

[4] Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation, 2023.

[5] Jiaxuan You, Rex Ying, Xiang Ren, William L. Hamilton, and Jure Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models, 2018.