HETEROGENEOUS MANIFOLDS FOR CURVATURE-AWARE GRAPH EMBEDDING

Francesco Di Giovanni[1]*, **Giulia Luise**[2]*, **Michael M. Bronstein**[1,3]
[1]Twitter [2]University College London [3]University of Oxford

ABSTRACT

The quality of graph embeddings depends on whether the geometry of the space matches that of the graph. Euclidean spaces are often a poor choice and recently hyperbolic spaces and more general manifolds, such as products of constant-curvature spaces and matrix manifolds, have resulted advantageous to better matching nodes pairwise distances. However, all these manifolds are *homogeneous*, implying that the curvature distribution is the same at each point, making them unsuited to match the local curvature (and related structural properties) of the graph. We study embeddings in a broader class of *heterogeneous* rotationally-symmetric manifolds. By adding a single radial dimension to existing homogeneous models, we can both account for heterogeneous curvature distributions on graphs and pairwise distances. We evaluate our approach on reconstruction tasks.

1 INTRODUCTION

Embedding data into a continuum space is at the heart of representation learning. For some time continuum was a synonym of Euclidean and it was usually assumed that data mapped to high-dimensional vectors likely lived on a smaller but *generally curved* submanifold. Recently a trend has emerged of encoding the geometry of the data *directly* into a richer ambient manifold. This approach has become popular for graph embeddings, since many real-world graphs exhibit power-law degree distribution and hierarchical structures typical of hyperbolic geometry Krioukov et al. (2010). It is thus not surprising that hyperbolic embeddings turned to be beneficial for reconstruction tasks such as link-prediction Nickel & Kiela (2017); Chamberlain et al. (2017). The improved performance is due to the space better matching structural properties of the input graph: this greater flexibility is encoded in the *curvature* information, contrarily to the flat Euclidean setting. Such findings have sparked interest in exploring different manifold classes such as products of constant curvature spaces (Gu et al. (2018), generalizing Wilson et al. (2014)), and matrix manifolds Cruceru et al. (2021), that could better accommodate the structural properties of graphs.

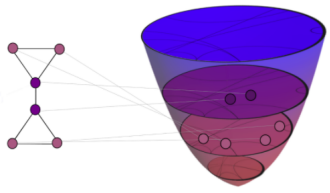


Figure 1: Left: nodes colored by discrete curvature. Right: a heterogeneous manifold proposed here, colored by its scalar curvature, and the graph-embedding.

These approaches have recently been noted to be instances of graph embeddings into symmetric manifolds López et al. (2021a). This family of manifolds are usually amenable to optimization techniques, partly because all points 'look the same', a feature known as *homogeneity*. Homogeneity often allows for closed and tractable formulas for distances and exponential maps that are required for Riemannian gradient descent algorithms Wilson & Leimeister (2018), but at the same time makes the ambient space 'stiff' since its curvature is position-independent. On the other hand, real-world graphs are often *heterogeneous*, where clustering and density generally vary from node to node. This local heterogeneous geometry can be encoded in discrete curvature Forman (2003); Ollivier (2007; 2009). We propose to go beyond the common strategy of minimizing distance-based losses and instead make our embedding *curvature-aware*, by also matching the node-wise curvature information with pointwise curvature on the manifold. This allows us to access structural information about the input graph from the *local properties of the manifold* rather than simply from the embedded nodes.

Contributions. In this paper, we propose what to our knowledge is the first method that preserves both distance and curvature information in graph representations. To this aim, we study a class of heterogeneous manifolds consisting of a product of a homogeneous factor and a spherically symmetric one. We show that this family generalize *any* homogeneous embedding studied so far and allow to both match the discrete curvature distribution and retain the computational tractability of standard approaches (something *generally lacking* on heterogeneous manifolds). We show that classical optimization techniques, such as Riemannian-SGD, extend to our heterogeneous spaces by computing a *single* additional derivative. We test this new approach on reconstruction tasks and conduct experiments that show its potential for triangles estimation and manifold random graphs.

Related work. Approximate isometric embeddings of graphs and similar objects have been extensively studied in theoretical geometry Gromov (1981); Linial et al. (1995); Indyk et al. (2017) and in computer graphics applications Mémoli & Sapiro (2005); Bronstein et al. (2006). The recent trend of 'geometric machine learning' Bronstein et al. (2017; 2021) attempts to incorporate geometric inductive biases such as symmetry and equivariance into deep neural networks and more broadly, leverage the geometric structure of the data and learning tasks. In the context of graphs, much of this research area is propelled by the success of graph neural networks Sperduti (1994); Goller & Kuchler (1996); Scarselli et al. (2008); Bruna et al. (2014), where the role of non-Euclidean geometry both in the form of representation of the node features Chami et al. (2019); Liu et al.; Bachmann et al. (2020) and tool to investigate limitations of existing architectures Chamberlain et al. (2021); Topping et al. (2021) is emerging. Our work continues the line of research of Wilson et al. (2014); Nickel & Kiela (2017); Gu et al. (2018); Cruceru et al. (2021); López et al. (2021a) and generalizes these methods by proposing a new class of ambient spaces and embedding algorithms.

2 Graph embedding in heterogeneous spaces

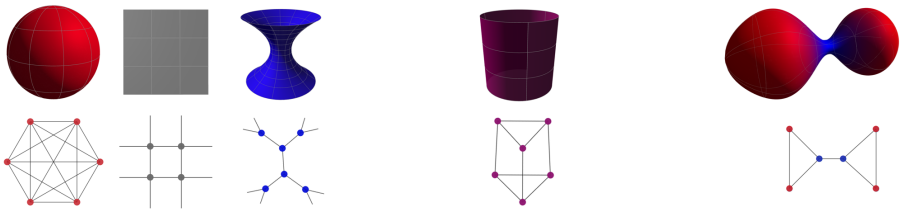
We refer to the appendix for notions of gradient, geodesics and exponential map on Riemannian manifolds. Below (M, g) is a Riemannian manifold with tangent space at p denoted by T_pM.

Homogeneous vs Heterogeneous spaces. For each point p in M, and for each pair of linearly independent tangent vectors $u, v \in T_pM$, the *sectional curvature* $K_p(u, v)$ at p is the Gaussian curvature (product of the minimal and maximal curvatures) of the surface spanned by $\exp_p(\{u, v\})$. When K_p is constant (in the sense that there exists $K \in \mathbb{R}$ such that $K_p(u, v) = K$ for any $p \in M$ and $u, v, \in T_pM$), then, up to quotients, M is either a sphere ($K > 0$), a Euclidean space ($K = 0$), or a hyperbolic space ($K < 0$). We refer to this special class as *space-forms*. More general than space-forms are *homogeneous manifolds*, which are characterized by the following property: for any points $p, q \in M$ there exists an isometry mapping p to q. This means that an observer cannot distinguish the point they are at based on the surrounding geometry. It follows that the sectional curvatures are only functions of the tangent vectors but *not* of the base-point, meaning that $K_p(\cdot, \cdot) = K(\cdot, \cdot)$ and hence that the scalar curvature - the simplest curvature term one can associate with a manifold (see Appendix) - is *constant*. Recent papers studied embedding graphs into homogeneous manifolds which hence cannot encode *any node-wise* information in their curvature being the latter constant Gu et al. (2018); Cruceru et al. (2021); López et al. (2021b). To overcome this rigidity one has to drop the homogeneity and consider more general manifolds, i.e. with non-constant (scalar) curvature, here referred to as *heterogeneous* (see Figure 2).

Curvature on graphs Although a graph does not come with a differential structure, synthetic notions of graph curvature have been introduced, most notably, by Forman (2003) and Ollivier (2007; 2009). In both cases, the idea is to consider an edge-based map that can recover some aspects of the Ricci curvature on manifolds, including relations to the volume growth rate Paeng (2012). Following Samal et al. (2018) we define the γ-*augmented Forman curvature* of an unweighted graph $G = (V, E)$ as the map $\mathrm{F} : E \to \mathbb{R}$

$$\mathrm{F}(i, j) = 4 - d_i - d_j + 3\gamma\, \sharp_\Delta(i, j), \quad \gamma > 0, \tag{1}$$

where d_i, d_j are the degrees of $i, j \in V$ respectively and $\sharp_\Delta(i, j)$ is the number of triangles based at the edge (i, j). We note that γ regulates the contribution of triangles and is generally set equal to one.



(a) Space forms (sphere, plane, hyperboloid)

(b) Product manifold

(c) Heterogeneous manifold

Figure 2: *Homogeneous* manifolds include (a) space-forms and (b) products of space-forms (e.g. cylinder) and have constant scalar curvature. A *heterogeneous* manifold (c) has non-constant scalar curvature and its graph counterpart has varying node-wise Forman curvature.

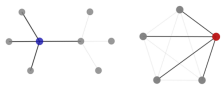


Figure 3: Nodes with same degree but different curvature.

As for manifolds, we can trace over the edges passing through i to compute the node-wise Forman scalar curvature $\mathrm{F}(i) = (\sum_{j:j\sim i} \mathrm{F}(i,j))/d_i$. We note that we can have nodes with same degrees but different (scalar) Forman curvatures describing distinct geometric configurations (Figure 3). If we are able to find an isometric embedding f of G into some space M, meaning that the average distance distortion AD_d vanishes, then we have preserved all the information and can thus fully reconstruct the graph. Distortions though are often unavoidable for a given ambient space Verbeek & Suri (2016). Partly motivated by these findings, we wish to leverage the structural information encoded in the Forman curvature to find embeddings that minimize AD_d *and* match the node-wise curvature of the graph with the point-wise curvature of the manifold. Accordingly, we need to consider embeddings into heterogeneous manifolds where the curvature information changes at each point, which we discuss next.

Spherically symmetric heterogeneous spaces The family of manifolds that we study are characterized by two features: a product structure and spherical symmetry. These two elements play a key role in ensuring that curvature and distance can be computed in *closed formulas*, something generally uncommon on manifolds. We refer to the appendix for full details on the construction.

Choose *any* homogeneous manifold (M_h, g_h) with a closed formula for the distance. Given a graph G with node set V and a manifold $M = M_h \times \mathbb{R}^3$, with M_h a homogeneous space and $\mathbb{R}^3$ a *rotationally symmetric space* as described in the appendix, we aim to find an embedding of the form

$$V \ni x \mapsto y(x) = (z(x), r(x), \bar{\theta}),$$

with $z(x) \in M_h$ and $(r(x), \bar{\theta})$ polar coordinates in $\mathbb{R}^3$ for some fixed angles $\bar{\theta} \in \mathbb{S}^2$. By doing so, the angles $\bar{\theta}$ enter neither the distance function nor the curvature one and are hence geometrically meaningless. Accordingly, we think of our embedding as simply adding a radial coordinate to our chosen homogeneous space $x \mapsto (z(x), r(x))$ to obtain a heterogeneous curvature now varying with r. We denote this class of embedding spaces by $M_h \times \mathcal{R}$. Note that our model generalizes any existing homogeneous one that in fact can be recovered by $(r = const)$.

Loss function. Similarly to Nickel & Kiela (2017); Gu et al. (2018); Cruceru et al. (2021), we construct embeddings by minimizing a suitable loss function. Thanks to our model we can minimize any distance and curvature depending loss via gradient descent. Let $V = \{x_1, \ldots, x_n\}$ and denote the embedded nodes by $\{y_i = (z_i, r_i)\}_{i=1}^n \subset M_h \times \mathcal{R}$. Using the shorthand $(y_1, \ldots, y_n) = \{y_i\}$, we consider a loss function of the form

$$\mathcal{L}(\{y_i\}) = \mathcal{L}_{\mathrm{d}}(\{y_i\}) + \tau \mathcal{L}_{\mathrm{c}}(\{y_i\}), \tag{2}$$

where τ is a scale parameter acting as a regularizer. We take $\mathcal{L}_d$ to be the average relative squared distance distortion (also known as 'dilation') and similarly for $\mathcal{L}_c$ (see Appendix for details). By minimizing $\mathcal{L}_d$ we account for long-range interactions in the form of pairwise distances meaning that we prioritize minimization of the average distance distortion AD_d. On the other hand, $\mathcal{L}_c$ is a curvature-based distortion so it measures *how close the local geometry of the manifold around the embedded nodes resembles that of the graph* G.

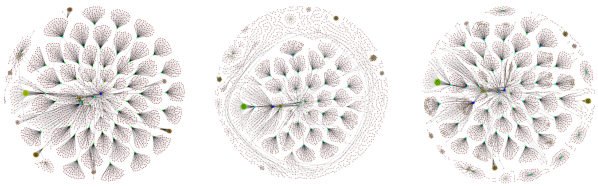


Figure 4: Web-edu graph (left) reconstructed using only distance in $\mathbb{H}^5 \times \mathbb{H}^5$ (center) and with a curvature-based correction (with 22.6% reconstruction improvement) in $\mathbb{H}^5 \times \mathbb{H}^5 \times \mathcal{R}$ (right).

Given our loss, we apply Riemannian stochastic gradient descent (R-SGD) Bonnabel (2013) to find an optimal embedding of a given graph G into our heterogeneous manifold. In the appendix we show that compared to the baseline homogeneous models, *the only additional term we need to compute for the gradient of $\mathcal{L}$ is a radial derivative*.

3 EXPERIMENTS

We experimentally evaluate graph embeddings into the new class of heterogeneous manifolds introduced in the previous section. We consider four real-world graphs: aves-wildbird Rossi & Ahmed (2015) (small animal network) CS-PhD De Nooy et al. (2018) (advisor-advisee relationship), web-edu Gleich et al. (2004) (web networks from the .edu domain) and Facebook Leskovec & Mcauley (2012) (dense social network). We show that our heterogeneous embeddings perform well w.r.t. distance-based metrics (average distance distortion AD_d and mean average precision mAP, see Appendix) while also matching the node-wise curvature information with the pointwise scalar curvature on the manifold. To assess the quality of the latter, we introduce the *average curvature distortion* AD_c (see Appendix). As baseline homogeneous embeddings, we use different products of space-forms M_h from Gu et al. (2018) and compare them to the heterogeneous embeddings constructed with the rotationally symmetric factor $M_h \times \mathcal{R}$. The results in Table 1 show that the proposed model attains reconstruction fidelity (in the sense of distance distortions) on par with the homogeneous baseline while also minimizing AD_c. In the homogeneous setting one can only match an *average* 'global curvature' as heuristically investigated in Gu et al. (2018) since the curvature is position-independent. Accordingly, computing AD_c is meaningless so we simply report the variance of the Forman curvature. In the appendix we also propose an investigation into better preservation of triangles via the curvature and how such improved fidelity can be exploited in the reconstruction (Figure 4). We also showcase the potential of our approach for generating heterogeneous random graphs from sampling on the family of rotationally symmetric manifolds introduced here.

Conclusions In this paper we proposed curvature-aware graph embeddings using a novel class of heterogeneous manifolds constructed as a product of a homogeneous space and a rotationally-symmetric manifold and offering a rich heterogeneous geometric structure together with computational tractability. Our approach extends all existing models for graph embeddings to faithfully approximate both distance-based and curvature-based information of the graph. We showed the effectiveness of the proposed approach on real-world graph reconstruction tasks and point towards the advantage of retaining curvature information in higher order structure detection and in controlling local behaviors when sampling random graphs.

Distance and Curvature Reconstruction Error												
$\lvert V\rvert$ / $\lvert E\rvert$	**Aves-Wildbird** 131 / 1444			**CS-PhD** 1025 / 1043			**WebEdu** 3031 / 6474			**Facebook** 4039 / 88324		
	AD_d	mAP	AD_c	AD_d	mAP	AD_c	AD_d	mAP	AD_c	AD_d	mAP	AD_c
$\mathbb{H}^5 \times \mathbb{H}^5$	.088	.99	(131)	.038	.96	(76.0)	.036	.98	(220)	.043	.77	($>10^4$)
$\mathbb{H}^5 \times \mathbb{H}^5 \times \mathcal{R}$	.085	.99	.001	.038	.99	.089	.036	.97	.018	.057	.76	.161
$\mathbb{H}^5 \times \mathbb{S}^5$	.090	.98	(131)	.050	.91	(76.0)	.050	.98	(220)	.059	.74	($>10^4$)
$\mathbb{H}^5 \times \mathbb{S}^5 \times \mathcal{R}$	.088	.98	.004	.043	.92	.186	.050	.99	.021	.076	.74	.163

Table 1: Perfomance of reconstruction tasks on real datasets. For homogeneous embeddings, AD_c reports the variance of Forman curvature.

REFERENCES

Martial Agueh and Guillaume Carlier. Barycenters in the Wasserstein space. *SIAM Journal on Mathematical Analysis*, 43(2):904–924, 2011.

Sigurd Angenent and Dan Knopf. An example of neckpinching for Ricci flow on S^{n+1}. *Mathematical Research Letters*, 11(4):493–518, 2004.

Gregor Bachmann, Gary Bécigneul, and Octavian Ganea. Constant curvature graph convolutional networks. In *International Conference on Machine Learning*, pp. 486–496. PMLR, 2020.

Federico Battiston, Giulia Cencetti, Iacopo Iacopini, Vito Latora, Maxime Lucas, Alice Patania, Jean-Gabriel Young, and Giovanni Petri. Networks beyond pairwise interactions: structure and dynamics. *Physics Reports*, 874:1–92, 2020.

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

Richard Bishop, Richard J Grittenden, et al. Geometry of manifolds. 1964.

Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013.

Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Sciences*, 103(5):1168–1172, 2006.

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017. URL `https://arxiv.org/abs/1611.08097`.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

Benjamin Chamberlain, James Rowbottom, Davide Eynard, Francesco Di Giovanni, Xiaowen Dong, and Michael Bronstein. Beltrami flow and neural diffusion on graphs. In *NeurIPS*, 2021.

Benjamin Paul Chamberlain, James Clough, and Marc Peter Deisenroth. Neural embeddings of graphs in hyperbolic space, 2017.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32:4868–4879, 2019.

Calin Cruceru, Gary Bécigneul, and Octavian-Eugen Ganea. Computationally tractable riemannian manifolds for graph embeddings. In *AAAI*, 2021.

Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. In *International conference on machine learning*, pp. 685–693. PMLR, 2014.

Wouter De Nooy, Andrej Mrvar, and Vladimir Batagelj. *Exploratory social network analysis with Pajek: Revised and expanded edition for updated software*, volume 46. Cambridge university press, 2018.

Francesco Di Giovanni. Rotationally symmetric Ricci flow on $\mathbb{R}^{n+1}$. *Advances in Mathematics*, 381:107621, 2021.

Robin Forman. Discrete and computational geometry, 2003.

David Gleich, Leonid Zhukov, and Pavel Berkhin. Fast parallel pagerank: A linear system approach. *Yahoo! Research Technical Report YRL-2004-038, available via http://research. yahoo. com/publication/YRL-2004-038. pdf*, 13:22, 2004.

Christoph Goller and Andreas Kuchler. Learning task-dependent distributed representations by backpropagation through structure. In *ICNN*, 1996.

Mikhael Gromov. Structures métriques pour les variétés riemanniennes. *Textes Mathématiques [Mathematical Texts]*, 1, 1981.

Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. Learning mixed-curvature representations in product spaces. In *International Conference on Learning Representations*, 2018.

Piotr Indyk, Jiří Matoušek, and Anastasios Sidiropoulos. 8: low-distortion embeddings of finite metric spaces. In *Handbook of discrete and computational geometry*, pp. 211–231. Chapman and Hall/CRC, 2017.

William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space 26. *Contemporary Mathematics*, 26, 1984.

Dmitri Krioukov, Fragkiskos Papadopoulos, Maksim Kitsak, Amin Vahdat, and Marián Boguná. Hyperbolic geometry of complex networks. *Physical Review E*, 82(3):036106, 2010.

Jure Leskovec and Julian Mcauley. Learning to discover social circles in ego networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/7a614fd06c325499f1680b9896beedeb-Paper.pdf.

Zhenhua Lin. Riemannian geometry of symmetric positive definite matrices via cholesky decomposition. *SIAM Journal on Matrix Analysis and Applications*, 40(4):1353–1370, 2019.

Nathan Linial, Eran London, and Yuri Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.

Qi Liu, Maximilian Nickel, and Douwe Kiela. Hyperbolic graph neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Federico López, Beatrice Pozzetti, Steve Trettel, Michael Strube, and Anna Wienhard. Symmetric spaces for graph embeddings: A finsler-riemannian approach. *arXiv preprint arXiv:2106.04941*, 2021a.

Federico López, Beatrice Pozzetti, Steve Trettel, Michael Strube, and Anna Wienhard. Vector-valued distance and gyrocalculus on the space of symmetric positive definite matrices. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021b.

Giulia Luise, Saverio Salzo, Massimiliano Pontil, and Carlo Ciliberto. Sinkhorn barycenters with free support via frank-wolfe algorithm. *Advances in Neural Information Processing Systems*, 32: 9322–9333, 2019.

Facundo Mémoli and Guillermo Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5(3):313–347, 2005.

Mark EJ Newman. Models of the small world. *Journal of Statistical Physics*, 101(3):819–841, 2000.

Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30:6338–6347, 2017.

Yann Ollivier. Ricci curvature of metric spaces. *Comptes Rendus Mathematique*, 345(11):643–646, 2007.

Yann Ollivier. Ricci curvature of markov chains on metric spaces. *Journal of Functional Analysis*, 256(3):810–864, 2009.

Seong-Hun Paeng. Volume and diameter of a graph and ollivier's ricci curvature. *European Journal of Combinatorics*, 33(8):1808–1819, 2012.

Peter Petersen. *Riemannian geometry*, volume 171. Springer, 2006.

Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL https://networkrepository.com.

Areejit Samal, RP Sreejith, Jiao Gu, Shiping Liu, Emil Saucan, and Jürgen Jost. Comparative analysis of two discretizations of ricci curvature for complex networks. *Scientific reports*, 8(1): 1–16, 2018.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

Aaron Sim, Maciej Wiatrak, Angus Brayne, Páidí Creed, and Saee Paliwal. Directed graph embeddings in pseudo-riemannian manifolds. *arXiv preprint arXiv:2106.08678*, 2021.

Alessandro Sperduti. Encoding labeled graphs by labeling RAAM. In *NIPS*, 1994.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. *arXiv preprint arXiv:2111.14522*, 2021.

Kevin Verbeek and Subhash Suri. Metric embedding, hyperbolic space, and social networks. *Computational Geometry*, 59:1–12, 2016. ISSN 0925-7721. doi: https://doi.org/10.1016/j.comgeo.2016.08.003. URL https://www.sciencedirect.com/science/article/pii/S0925772116300712.

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

Benjamin Wilson and Matthias Leimeister. Gradient descent in hyperbolic space. *arXiv preprint arXiv:1805.08207*, 2018.

Richard C Wilson, Edwin R Hancock, Elżbieta Pekalska, and Robert PW Duin. Spherical and hyperbolic embeddings of data. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2255–2269, 2014.

A EMBEDDING INTO RIEMANNIAN MANIFOLDS

The setting. We consider an undirected graph $G = (V, E)$ with n nodes. For $i \in V$ we let $\mathcal{N}_i$ be the *neighbourhood* of i and $d_i = |\mathcal{N}_i|$ the *degree* of i. The *(geodesic) distance*[1] $d_G(i, j)$ is the length of the shortest walk connecting nodes i and j. We are interested in finding a target space M equipped with metric d_M and an embedding $f : V \to M$ so that the graph can be *reconstructed* from f and M. If there exists an *isometric* map $f : V \to M$, i.e. satisfying $d_M(f(i), f(j)) = d_G(i, j)$ for all $i, j \in V$, then we can perfectly reconstruct the input data. However, given constraints on M such as bounded dimension, a perfect isometric embedding is typically unavailable, and one tries to find a 'least-distorting' embedding, in some sense: the *average (distance) distortion* $\mathsf{AD}_d(f)$

$$\mathsf{AD}_d(f) := \frac{2}{n(n-1)} \sum_{i,j=1}^{n} \left| 1 - \frac{d_M(f(i), f(j))}{d_G(i, j)} \right| \tag{3}$$

and the *mean average precision* mAP

$$\mathsf{mAP}(f) := \frac{1}{n} \sum_{i \in V} \frac{1}{d_i} \sum_{j \in \mathcal{N}_i} \frac{|\mathcal{N}_i \cap \mathcal{R}_{i,j}|}{|\mathcal{R}_{i,j}|}, \tag{4}$$

are two common criteria, where $\mathcal{R}_{i,j}$ is the set of nodes $z \in V$ such that $d_M(f(i), f(z)) \leq d_M(f(i), f(j))$. We note that while $\mathsf{AD}_d(f)$ is affected by pairwise distances beyond the 1-hop neighbourhood, the mAP is a measure of how well an embedding is able to reproduce the 1-hop neighbourhood of a node disregarding the actual scale of distances. We refer to Section 4 in Cruceru et al. (2021) for a thorough discussion on the matter of choosing the right criterion for embedding distortion.

In general, distortions are inevitable and tend to accumulate on higher-order structures Verbeek & Suri (2016), which are important in many practical applications such as social networks Benson et al. (2016) and physical systems Battiston et al. (2020). In this case, it is desirable to go beyond pairwise distances and access other types of information on the ambient space to better reconstruct the input data. Discrete graph curvature Forman (2003); Ollivier (2007) is one way of accounting for such structures. In the rest of the paper, we study embeddings that can both minimize AD_d (or maximize mAP) and account for local graph structural properties by matching the graph curvature distribution with that of a suitable class of target spaces.

A.1 RIEMANNIAN MANIFOLDS

A natural class of continuous embedding spaces for graphs are Riemannian manifolds Petersen (2006), since they come with a differentiable structure and are hence amenable to optimization methods. Informally, a d-dimensional manifold M is a topological space that can be locally identified with Euclidean space via smooth maps: hence for every point $p \in M$ there exists an associated *tangent space* $T_pM \cong \mathbb{R}^d$. Assume we are given a positive-definite inner product $g_p : T_pM \times T_pM \to \mathbb{R}$. If the assignment $p \mapsto g_p$ is smoothly compatible with the differentiable structure of M, we refer to g as a ***Riemannian metric (tensor)*** on M.

Geodesics. The Riemannian metric g induces a distance function d_g that measures the length of minimal paths on the manifold M called *geodesics*. An important property of the distance is that $d_g^2(\cdot, p)$ is smooth locally[2] around p, meaning that any loss $\mathcal{L}$ depending on d_g^2 is locally smooth on M and can hence be optimized by first order methods.

Exponential map. Given $v \in T_pM$, the *exponential map* $\exp_p : T_pM \to M$ yields the point in M obtained by travelling for unit time along the geodesic starting at p with initial speed v. The exponential map plays a key role in optimization on manifolds, allowing to update an embedding at p based on gradients of the loss living in the tangent space.

[1] Distances are often termed 'metrics'. Here, we prefer the term 'distance' to avoid confusion with Riemannian metrics.

[2] Namely, away from the cut-locus Petersen (2006).

Embedding. The problem of isometric (metric-preserving) embeddings of discrete metric spaces (and graph in particular) has been extensively studied both in theoretical and applied literature Linial et al. (1995); Indyk et al. (2017); Johnson & Lindenstrauss (1984). In general, a graph cannot be isometrically embedded into a fixed space; the structure and the dimension of the embedding space have a crucial effect on the embedding distortion. Typically, increasing the dimension of the space allows to reduce the distortion, however, it comes at the expense of memory and computational cost. For this reason, one often seeks a lower-dimensional space with 'richer' structure that is better suited for the graph. When using Riemannian manifolds for graph embeddings, the 'richness' of the space M is manifested in its *curvature*, which we define next.

Curvature. For each point p in M, and for each pair of linearly independent tangent vectors $u, v \in T_pM$, the *sectional curvature* $K_p(u, v)$ at p is the Gaussian curvature (product of the minimal and maximal curvatures) of the surface spanned by $\exp_p(\{u, v\})$. Given a tangent vector v at p, if we 'average' the sectional curvatures at p over a set of orthonormal vectors we obtain a bilinear form $\mathrm{Ric}_p : T_pM \times T_pM \to \mathbb{R}$ called *Ricci curvature*. This bilinear map is related to the volume growth rate and the propagation of information Petersen (2006)[Chapter 9]. By computing the trace of Ric, we finally obtain a map $\mathrm{R} : M \to \mathbb{R}$ called *scalar curvature*. This is the simplest curvature term one can associate with a manifold and the most natural quantity to adopt when fitting the node-wise curvature on a graph.

Gradient on manifolds. If (M, g) is a Riemannian manifold, a vector field X is a smooth map $X : p \mapsto X_p \in T_pM$ assigning to each point in M a tangent vector. We recall that tangent vectors on a manifold represent linear differential operators, meaning that for any smooth function $f : M \to \mathbb{R}$ and vector field X, we can construct a smooth function $X(f) : p \mapsto X_p(f) = df_p(X_p)$.

Given $f : M \to \mathbb{R}$ smooth, the gradient of f with respect to g is the vector field $\nabla_g f$ satisfying $df(X)|_p = g_p(X, \nabla_g f)$ for any vector field X. Given local coordinates $\{x^i\}$ around $p \in M$, we can express the gradient of f as

$$(\nabla_g f)|_p^i = \sum_j (g_p^{-1})_{ij} \partial_j f(p). \tag{5}$$

Therefore, if we have a loss $\mathcal{L}$, once we compute its Euclidean gradient with respect to coordinates on M and hence obtain the tangent vector $(\partial_1 \mathcal{L}, \dots, \partial_d \mathcal{L})$, we need to further project it using the inverse metric g^{-1}.

Since many real graphs are characterized by features as power-law degree distribution and hierarchical structures that are intrinsic to hyperbolic geometry Krioukov et al. (2010), in most of our evaluations we let the homogeneous factor be a product of space forms containing at least one hyperbolic term. Accordingly, we first briefly review the hyperboloid model we adopt following the discussion in Wilson & Leimeister (2018).

Hyperboloid model. We consider the Minkowski product on $\mathbb{R}^{d+1}$ defined by

$$\langle x, y \rangle_{d:1} := \sum_{i=1}^{d} x_i y_i - x_{d+1} y_{d+1},$$

with signature $(+, \dots, +, -)$. The d-dimensional hyperbolic space $\mathbb{H}^d$ can be described as

$$\mathbb{H}^d = \left\{ x \in \mathbb{R}^{d+1} : \langle x, x \rangle_{d:1} = -1, x_{d+1} > 0 \right\}.$$

The distance between two points on the hyperbolic space is given by

$$d_{\mathbb{H}^d}(x, y) = \operatorname{arcosh}(-\langle x, y \rangle_{d:1}).$$

The tangent space $T_p\mathbb{H}^d$ is isomorphic to the null directions with respect to the Minkowski product at p. Finally, the exponential map is given by

$$\exp_p(v) = \cosh(||v||)p + \sinh(||v||)\frac{v}{||v||}.$$

Assume now we have a distance-based loss $\mathcal{L}_d$ defined on the hyperboloid model. One first has to compute the standard gradient $\nabla \mathcal{L}$ in the ambient space $\mathbb{R}^{d+1}$. Then, in light of equation 5 we rescale using the (inverse) Minkowski metric deriving

$$\nabla^{d:1} \mathcal{L} = \mathrm{diag}(+, \dots, +, -) \nabla \mathcal{L}.$$

Finally we project $\nabla^{d:1}$ to the tangent space of the hyperboloid to derive $\nabla^{\mathbb{H}^d} \mathcal{L}$.

Homogeneity of existing models. In this paragraph we briefly review that the main existing models analysed recently in Nickel & Kiela (2017); Gu et al. (2018); Cruceru et al. (2021); López et al. (2021a) are indeed homogeneous and hence their curvature information cannot generally match the graph discrete one. In fact, as observed in López et al. (2021a), the homogeneity follows from the stronger requirement of *symmetry* Petersen (2006)[Chapter 8.1]: symmetric spaces are characterized by the property that for each point p there exists an isometry fixing p whose Jacobian is minus the identity.

In the case of graph embeddings into space forms Wilson et al. (2014); Nickel & Kiela (2017), the homogeneity is a known fact and follows for example from the curvature tensor being covariantly constant. This extends to the cartesian products of such spaces analysed in Gu et al. (2018). In Cruceru et al. (2021) two Riemannian manifolds have been investigated regarding the graph embedding problem: the SPD manifold (i.e. symmetric positive definite matrices) and the Grassmannian manifold. The homogeneity is a consequence of their Lie group structure (see Lin (2019) and Petersen (2006)[Chapter 8.2] respectively). Finally another class of symmetric spaces - Siegel manifolds - have been studied in López et al. (2021a). In particular, we refer to the Appendix of López et al. (2021a) for a more detailed discussion of symmetric manifolds and why they are generally advantageous in representation learning tasks López et al. (2021b).

As observed in Section 2, since the curvature of a Riemannian metric g is invariant under isometries, when g is homogeneous - meaning that its isometry group acts *transitively* - the curvature of g cannot distinguish between two given points p, q otherwise it would break the invariance with respect to the isometry mapping p to q. This key property of homogeneous spaces is arguably the main reason why they appear so frequently whenever optimization on manifolds is required: the position-independence of the curvatures makes the geometry of the space the same around each point, often leading to closed and tractable formulas for the distance function and exponential maps, which are generally unavailable. *On the other hand, this rigidity comes with a price: no information about the input graph domain can be derived from the underlying continuum space and its geometry without reconstructing an adjacency skeleton on the embedded point cloud.*

This last point is at the heart of our work, where we are interested in encoding the geometry of the data in the actual continuum texture of the ambient space via curvature matching. To this aim, one has to consider heterogeneous manifolds with position aware curvature.

B A TRACTABLE CLASS OF HETEROGENEOUS MANIFOLDS

We first rewiew the two main apsects of our construction: product structure and rotational symmetry. We also note that in the following we use spherical and rotational interchangeably.

Product manifolds. Given two Riemannian manifolds (M_1, g_1) and (M_2, g_2), their Cartesian product $M := M_1 \times M_2$ can be equipped with a standard Riemannian structure $g := g_1 \oplus g_2$. The product structure allows to easily compute relevant quantities such as distance, exponential map and scalar curvature from each factor. The squared distance function on (M, g) and the exponential map are given by

$$d_g^2((p_1, p_2), (q_1, q_2)) = d_{g_1}^2(p_1, q_1) + d_{g_2}^2(p_2, q_2), \tag{6}$$

$$\exp_g |_{(p_1,p_2)}(X) = (\exp_{g_1} |_{p_1}(X_1), \exp_{g_2} |_{p_2}(X_2)), \tag{7}$$

respectively, where $p = (p_1, p_2)$ and $q = (q_1, q_2)$ are points on M. The scalar curvature is simply given by the sum

$$\mathrm{R}_g(p_1, p_2) = \mathrm{R}_{g_1}(p_1) + \mathrm{R}_{g_2}(p_2). \tag{8}$$

The above decomposition shows that for optimizing a distance and curvature dependent loss on a product manifold it suffices to follow the Riemannian gradient descent *on each factor separately*.

Rotationally symmetric manifolds. Consider polar coordinates $\{(r, \theta, \psi)\}$ in $\mathbb{R}^3$. We can write the Euclidean metric in such coordinates as $g_E = dr^2 + r^2(d\theta^2 + \sin^2(\theta)d\psi^2) = dr^2 + r^2 g_{\mathbb{S}^2}$, where $g_{\mathbb{S}^2}$ is the standard metric on the 2-sphere. We can generalize this construction to the class of metrics that are invariant under rotations and can hence be written in polar coordinates as

$$g_\varphi = dr^2 + \varphi^2(r) g_{\mathbb{S}^2}, \tag{9}$$

for some smooth function φ. In fact, we note that a generally rotationally symmetric metric g_φ on $(0,\infty)\times\mathbb{S}^m$ can be written as

$$g_\varphi = dr^2 + \varphi^2(r) g_{\mathbb{S}^m}$$

up to renaming r to be the coordinate representing the distance from the singular orbit (origin in our case), as for example in Angenent & Knopf (2004); Di Giovanni (2021). The metric g_φ on the open manifold $(0,+\infty)\times\mathbb{S}^m$ defines a smooth complete metric on $\mathbb{R}^{m+1}$ if φ extends to an odd function at the origin with $\partial_r\varphi(0)=1$ Petersen (2006)[Chapter 1.4]. It is worth emphasizing that the choice $\varphi(r)=\sinh(r)$ recover the Hyperbolic space as well.

The curvature information of a spherically symmetric metric on $\mathbb{R}^{m+1}$ is encoded in the sectional curvature K of the 2-plane perpendicular to the spherical orbits and L of the 2-plane tangential to such orbits Petersen (2006)[Chapter 3.2]:

$$K = -\frac{\partial^2_{rr}\varphi}{\varphi}, \quad L = \frac{1-(\partial_r\varphi)^2}{\varphi^2}. \tag{10}$$

By tracing we derive the formulas for the Ricci curvature and the scalar curvature:

$$\mathrm{Ric} = -m\frac{\partial^2_{rr}\varphi}{\varphi}dr^2 + (-\varphi\,\partial_{rr}\varphi + (m-1)(1-(\partial_r\varphi)^2))g_{\mathbb{S}^m}, \tag{11}$$

$$\mathrm{R} = m\left(-2\frac{\partial^2_{rr}\varphi}{\varphi} + (m-1)\frac{1-(\partial_r\varphi)^2}{\varphi^2}\right) \tag{12}$$

In particular in the three-dimensional case, i.e. $m=2$, we have

$$\mathrm{R}(r) = 2\left(\frac{-2\partial^2_{rr}\varphi}{\varphi} + \frac{1-(\partial_r\varphi)^2}{\varphi^2}\right)(r). \tag{13}$$

The curvature depends on the radial coordinate r, meaning that $r\mapsto \mathrm{R}(r)$ is *non-constant* and hence that the resulting space is *heterogeneous* (except for very particular choices of φ). We emphasize how the curvature is instead independent of angular coordinates θ,ψ.

We also note that given two points p,q lying along the same ray, i.e. $p=(r_0,\bar\theta,\bar\psi)$ and $q=(r_1,\bar\theta,\bar\psi)$ for some angles $\bar\theta,\bar\psi$, then we have a simple formula for their distance:

$$d_g(p,q) = |r_1 - r_0|. \tag{14}$$

In the following, we pick one specific instance of rotationally symmetric space, by choosing the radial function φ in equation 9 of the form

$$\varphi_\alpha(r) = \alpha\arctan\left(\frac{r}{\alpha}\right), \quad \alpha>0. \tag{15}$$

While other options are viable, this choice is motivated by some convenient properties of the metric in equation 9. The resulting geometry resembles a hemisphere glued to a cylinder with positive monotone decreasing scalar curvature while α controls the radius of the hemisphere and hence how curved the space is (Figure 5). In practice, the value α depends on two tunable hyperparameters that allow us to vary the range of curvatures on the manifold to match the discrete curvature of the embedded graph as detailed below. Alternatively, α - or more generally φ - could be learned based on the problem.

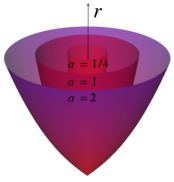


Figure 5: Visualisation of 3D rotationally symmetric manifolds for different α.

Details on our explicit choice. It follows from equation 10 that any smooth concave function φ with $\partial_r\varphi(0)=1$ gives rise to a spherically symmetric metric with nonnegative sectional curvature $K\geq 0$, $L\geq 0$ and hence $\mathrm{R}\geq 0$. The concavity and monotonicity of the warping function φ characterizes strongly the geometry of these spaces and indeed has an impact for example on the type of singularities that the Ricci flow may develop Angenent & Knopf (2004); Di Giovanni (2021).

In particular our choice in equation 15 reported below

$$\varphi_\alpha : r \mapsto \varphi_\alpha(r) = \alpha \arctan\left(\frac{r}{\alpha}\right), \quad \alpha > 0$$

satisfies these properties.

Since φ is odd with $\partial_r \varphi_\alpha(0) = 1$, the manifold $(\mathbb{R}^{m+1}, g_{\varphi_\alpha})$ is smooth and complete. Moreover, as mentioned above $\mathrm{R}_\alpha \geq 0$. In fact, a standard de l'Hôpital argument gives

$$\mathrm{R}_\alpha(0) = \frac{12}{\alpha^2}.$$

By direct computation one can also check that $\partial_r \mathrm{R}_\alpha \leq 0$ with

$$\inf_{r \geq 0} \mathrm{R}_\alpha(r) = \frac{8}{\alpha^2 \pi^2}.$$

Note how this is not surprising, since geometrically the manifold looks like a round cylinder away from the origin. In particular, we see that α affects the range of curvature and how positively curved the manifold is at the origin as illustrated in Figure 5. We will see below how to choose α based on the range of curvatures on the given input graph we want to match.

The role of rescaling. We emphasize how the curvature *monotonicity* is in general a feature helping the fitting of the Forman distribution on the graph since otherwise the gradient of the curvature-based loss $\mathcal{L}_c$ could get stuck at one stationary point of R_α. In fact, this property also allows us to have a better control and interpretability of the hyperparameters entering the model as discussed in the next section. On the other hand, since the graph is still a non-differentiable structure, it may happen that adjacent nodes have highly varying Forman curvature: equivalently, the node-wise Forman curvature may have large Dirichlet energy

$$E(\mathrm{F}) = \frac{1}{2} \sum_{i \sim j} \left\| \frac{\mathrm{F}_i}{\sqrt{d_i}} - \frac{\mathrm{F}_j}{\sqrt{d_j}} \right\|^2.$$

Accordingly, to avoid sacrificing the distance-based loss, we grant the model an extra degree of freedom given by a rescaling λ^2 of the rotationally symmetric metric g_φ. We usually take λ to be a contraction, meaning that the projection of the distance function on the radial directions in equation 14 becomes $\lambda d_{g_\varphi}((r_0, \cdot), (r_1, \cdot))$ hence allowing us to weight less the rotationally symmetric space in the distance-based loss $\mathcal{L}_d$. On the other hand the scalar curvature of the rescaled metric transforms as $\mathrm{R}_\alpha / \lambda^2$. In the next section we will describe how to deal with the curvature rescaling and in general make the curvature matching component of our approach more robust to both the choice of the homogeneous factor and of the rescaling factor.

Volume growth measurement Here we describe how we accounted for the volume on the synthetic reconstruction task in Figure 6. We choose our homogeneous space to be the standard 3-dimensional Hyperbolic space $(M_h, g_h) = (\mathbb{H}^3, g_{\mathbb{H}^3})$.

Since our embedding is spherically symmetric, instead of considering geodesic balls $B_g(y_i, \rho)$ in our heterogeneous space $\mathbb{H}^3 \times \mathcal{R}$, we look at annular regions

$$V(y_i, \rho) := \{y = (z, r, \theta) : \ d^2_{\mathbb{H}^3}(z, z_i) + (r - r_i)^2 < \rho^2\},$$

so that we can explicitly compute volumes and exploit the fact that our geometry (e.g. ordering of nodes with larger volumes at a given radius) is independent of the angular coordinates. We then have

$$v(y_i, \rho) := \mathrm{vol}_g(V(y_i, \rho)) = \int_{(r_i - \rho)_+}^{r_i + \rho} |B_{g_{\mathbb{H}^3}}(o, \sqrt{\rho^2 - (r - r_i)^2})|_{g_{\mathbb{H}^3}} \, \omega_2 \, \varphi^2_\alpha(r) dr,$$

with ω_2 the area of the 2-sphere and o some fixed point of our hyperbolic model. Say we consider the rotationally symmetric model for the three-dimensional hyperbolic space, then

$$v(y_i, \rho) = (\omega_2)^2 \int_{(r_i - \rho)_+}^{r_i + \rho} \left(\int_0^{\sqrt{\rho^2 - (r - r_i)^2}} \sinh^2(z) dz \right) \varphi^2_\alpha(r) dr. \tag{16}$$

This is the quantity we use to match the volume reconstruction of the graph as in Figure 6.

B.1 TRACTABLE HETEROGENEOUS MANIFOLDS.

Choose *any* homogeneous manifold (M_h, g_h) with a closed formula for the distance and consider a rotationally symmetric space (e.g. with φ as in equation 15). We build a heterogeneous space as the product $M_h \times \mathbb{R}^3$ equipped with the metric $g_h \oplus g_\varphi$.

We now describe how one can effectively rely on the structure of the heterogeneous manifolds introduced above to learn curvature aware graph representations (see Figure 1).
Given a graph G with node set V and a manifold $M = M_h \times \mathbb{R}^3$, with M_h a homogeneous space, we aim to find an embedding of the form

$$V \ni x \mapsto y(x) = (z(x), r(x), \bar{\theta}),$$

with $z(x) \in M_h$ and $(r(x), \bar{\theta})$ polar coordinates in $\mathbb{R}^3$ for some fixed angles $\bar{\theta} \in \mathbb{S}^2$. According to equation 6 and equation 14, the squared distance between two points $y_0 = (z_0, r_0, \bar{\theta})$ and $y_1 = (z_1, r_1, \bar{\theta})$ is

$$d_g^2((z_0, r_0, \bar{\theta}), (z_1, r_1, \bar{\theta})) = d_h^2(z_0, z_1) + (r_1 - r_0)^2, \tag{17}$$

where d_h is the distance on the homogeneous factor. On the other hand, from equation 8 we derive that for any $(z, r, \bar{\theta})$ in M the following holds:

$$\mathrm{R}_g(z, r, \bar{\theta}) = \mathrm{R}_{g_h}(z) + \mathrm{R}_\alpha(r) = \mathrm{R}_h + \mathrm{R}_\alpha(r) \tag{18}$$

where we have used that the scalar curvature of the homogeneous factor R_h is a constant and $\mathrm{R}_\alpha(r)$ is the scalar curvature of the rotationally symmetric factor given by equation 13 with φ_α as in equation 15. Note that by embedding the nodes along a ray in the rotationally symmetric factor, the angles enter neither the distance function nor the curvature one and are hence geometrically meaningless. Accordingly, we think of our embedding as simply adding a radial coordinate to our chosen homogeneous space $x \mapsto (z(x), r(x))$ to obtain a heterogeneous curvature now varying with r. Therefore, we **simplify our notation** and denote this class of embedding spaces by $M_h \times \mathcal{R}$ to emphasize that *there is only one additional dimension compared to the homogeneous baseline*. From now on, we tacitly assume that any heterogeneous graph embedding is of this form. Note in particular that if we embed the graph into $r = \text{const}$, we recover existing homogeneous models. In the following, we usually take (M_h, g_h) to either be a space-form or a product thereof as in Gu et al. (2018).

Example. Let $\mathbb{H}^d$ denote the standard d-dimensional hyperbolic space. Since the scalar curvature of $\mathbb{H}^d$ is given by $\mathrm{R}_{\mathbb{H}^d} = -d(d-1)$, if we consider for example an embedding $x \mapsto (z(x), r(x)) \in \mathbb{H}^5 \times \mathbb{H}^5 \times \mathcal{R}$, then equation 18 becomes $\mathrm{R}(z(x), r(x)) = 2\,(-5 \cdot 4) + \mathrm{R}_\alpha(r(x))$.

C DETAILS ON THE ALGORITHM

Loss function. Similarly to Nickel & Kiela (2017); Gu et al. (2018); Cruceru et al. (2021), we construct embeddings by minimizing a suitable loss function. Thanks to equation 17 and equation 18 we can minimize any distance and curvature depending loss via gradient descent. Let $V = \{x_1, \ldots, x_n\}$ and denote the embedded nodes by $\{y_i = (z_i, r_i)\}_{i=1}^n \subset M_h \times \mathcal{R}$. Using the shorthand $(y_1, \ldots, y_n) = \{y_i\}$, we consider a loss function of the form

$$\mathcal{L}(\{y_i\}) = \mathcal{L}_{\mathrm{d}}(\{y_i\}) + \tau \mathcal{L}_{\mathrm{c}}(\{y_i\}), \tag{19}$$

where τ is a scale parameter acting as a regularizer. We take $\mathcal{L}_d$ to be the average relative squared distance distortion (also known as 'dilation')

$$\mathcal{L}_d(\{y_i\}) = \sum_{i,j} \left| \frac{d_{g_h}^2(z_i, z_j) + (r_i - r_j)^2}{d_G^2(x_i, x_j)} - 1 \right| \tag{20}$$

where we have used equation 17 to compute the squared distance of embedded nodes. This has the advantage of depending only on the squared distance functions which we recall to be locally smooth. On the other hand, $\mathcal{L}_c$ is a new curvature-based loss

$$\mathcal{L}_c(\{y_i\}) = \sum_i \frac{\left(\mathrm{F}(x_i) - \mathrm{R}_h - \mathrm{R}_\alpha(r_i)\right)^2}{\left(|\mathrm{F}(x_i)| + \epsilon\right)^2}, \tag{21}$$

where ϵ is a constant to avoid numerical instabilities. Note that this is just one option to encourage curvature matching and other losses can be adopted.

C.1 ALGORITHM

Given our loss in equation 19, we apply Riemannian stochastic gradient descent (R-SGD) Bonnabel (2013) to find an optimal embedding of a given graph G into our heterogeneous manifold. Assume that we have chosen a homogeneous factor M_h and we have mapped the nodes to the points $(z_i, r_i) \in M_h \times \mathcal{R}$. We denote by U the update on the homogeneous coordinates $\{z_i^{(t)}\}$ based on a step of R-SGD of the loss in equation 19. We show the following:

Proposition C.1. *If we apply R-SGD to $\mathcal{L} = \mathcal{L}_d + \tau\mathcal{L}_c$, the update of the radial component simply becomes:*

$$(r_i^{(t)} + \partial_{r_i}\mathcal{L})_+ \leftarrow r_i^{(t)}, \tag{22}$$

with $(\cdot)_+$ the positive part. Therefore, the update on the product space is

$$(\mathsf{U}(z_i^{(t)}), (r_i^{(t)} + \partial_{r_i}\mathcal{L})_+)) \leftarrow (z_i^{(t)}, r_i^{(t)}). \tag{23}$$

Proof of Proposition C.1. Let (M_h, g_h) be a chosen homogeneous manifold. Suppose we have $f : M \to \mathbb{R}$ smooth with $M = M_h \times \mathcal{R}$, meaning that we consider the heterogeneous manifold $(M, g) = (M_h, g_h) \times (\mathbb{R}^3, g_\varphi)$, with g_φ as in equation 9 for some smooth radial map φ. Assume that f is $SO(3)$-invariant, i.e. that given $z \in M_h$ and $r > 0$ we have

$$f(z, r, \theta_0) = f(z, r, \theta_1), \quad \forall \theta_0, \theta_1 \in \mathbb{S}^2.$$

We note that this is the case for our loss $\mathcal{L}$ in equation 19 which is independent of angular coordinates. Since g is a product metric the tangent space of M is the direct sum of the tangent spaces of the individual factors and we can write the gradient of f as

$$\nabla_g f(z_0, r_0, \theta_0) = \left(\nabla_{g_h} f(\cdot, r_0, \theta_0)\right)(z_0) \oplus \left(\nabla_{g_\varphi} f(z_0, \cdot, \cdot)\right)(r_0, \theta_0).$$

Since f is spherically symmetric, the right hand side becomes

$$\nabla_g f(z_0, r_0, \theta_0) = \left(\nabla_{g_h} f(\cdot, r_0, \theta_0)\right)(z_0) \oplus \partial_r f(z_0, \cdot, \theta_0)(r_0)\partial_r,$$

where we have also used equation 5 and that the inverse metric g_φ^{-1} writes as

$$g_\varphi^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & g_{\mathbb{S}^2}^{-1} \end{pmatrix}.$$

On the other hand, if $\nu \in \mathbb{R}$, the unique g_φ-radial geodesic starting at some (r_0, θ_0) with initial tangent vector $\nu\partial_r$ is

$$t \mapsto (r_0 + t\nu, \theta_0),$$

meaning that the exponential map is always defined at $\nu\partial_r$ and is given explicitly by

$$(\exp_{g_\varphi})|_{(r_0,\theta_0)}(\nu\partial_r) = ((r_0 + \nu)_+, \theta_0).$$

Therefore, we may apply equation 7 and conclude that

$$(\exp_g)_{(z_0,r_0,\theta_0)}(\nabla_g f) = \left((\exp_{g_h})|_{z_0}(\nabla_{g_h} f(\cdot, r_0, \theta_0)(z_0)), (r_0 + (\partial_r f(z_0, \cdot, \theta_0))(r_0))_+, \theta_0\right).$$

If we apply the previous computation to each component of $\mathcal{L}$, we then get the update rule for the R-SGD algorithm. □

We see that compared to the baseline homogeneous models, the only additional term we need to compute for the gradient of $\mathcal{L}$ is a radial derivative.

Our construction generalizes to *weighted* products of homogeneous factors and rotationally symmetric spaces. Specifically, this means considering $M_h \times \lambda^2\mathcal{R}$ with $\lambda > 0$. The scaling λ impacts equation 17 and equation 18 and results in weighting less the radial contribution to the distance function effectively allowing us to match the node-wise graph curvature with the pointwise continuous one more easily. Since scaling the curvature generally affects the range of curvatures, we can also allow the curvature matching to be up to a linear transformation. This is discussed further in the next paragraph.

Matching curvature up to invertible linear maps Next, we discuss how we allowed our algorithm to have an extra degree of freedom to more easily account for distances and curvatures simultaneously. In general the curvature of the heterogeneous model $M_h \times \lambda^2 \mathcal{R}$ writes as

$$\mathrm{R} = \mathrm{R}_h + \frac{1}{\lambda^2}\mathrm{R}_\alpha$$

where we are using the explicit choice in equation 15 for the spherically symmetric factor with $\lambda > 0$ a positive rescaling on the rotationally symmetric space introduced above. In general, we wish to make our model robust with respect to the choice of the homogeneous factor given that it only leads to a constant value R_h translating the global curvature of our ambient space. Similarly, the role of the rescaling should not affect how we match the node-wise curvature distribution. Accordingly, we propose to reconstruct the curvature information at node x_i by the curvature on the manifold at the embedded point $y_i = (z_i, r_i)$ up to a known shifting and rescaling. It means that for our curvature-based loss we instead minimize

$$\mathcal{L}_c(y_1, \dots, y_n) = \sum_i \left(\mathrm{F}(x_i) - \lambda^2 \left(\mathrm{R}_h + \frac{\mathrm{R}_\alpha(r_i)}{\lambda^2} \right) + \rho \right)^2, \tag{24}$$

where we take the translation ρ of the form

$$\rho = \lambda^2 \mathrm{R}_h - \min \mathrm{F} + \hat{\delta}_\alpha,$$

with $\min \mathrm{F}$ the minimum of scalar Forman on the given graph G and $\hat{\delta}_\alpha$ a constant we discuss in the next section. Therefore, equation 24 becomes

$$\mathcal{L}_c(y_1, \dots, y_n) = \sum_i \left(\mathrm{F}(x_i) - \mathrm{R}_\alpha(r_i) - \min \mathrm{F} + \hat{\delta}_\alpha \right)^2 \tag{25}$$

making it **independent** of the choice of the homogeneous factor M_h. It remains to discuss the role of $\hat{\delta}_\alpha$ and in general how by using the geometry of the problem we can tune two hyperparameters to ensure the curvature matching.

C.2 Tuning geometric hyperparameters

To allow the manifold scalar curvature to fit the node-wise Forman signal, we see from equation 25 that a necessary requirement is considering range of curvatures that cover the interval $\max \mathrm{F} - \min \mathrm{F}$. Since by choice R_α is *monotone*, we immediately see that the $\mathrm{argmin}\mathrm{F}$ on the graph should be mapped to the radial coordinate $r_{\min}$ where $\mathrm{R}_\alpha(r_{\min}) = \hat{\delta}_\alpha$. We know that given α, the function R_α admits a horizontal asymptote given by $8/(\pi\alpha)^2$, therefore we find the constraint

$$\hat{\delta}_\alpha \geq \frac{8}{\pi^2 \alpha^2}. \tag{26}$$

On the other hand, a symmetric argument works for $\max \mathrm{F}$: in fact, if we denote by $r_{\max}$ the radial coordinate we should map $\mathrm{argmax}\, \mathrm{F}$ to, by monotonicity we have the constraint

$$\frac{12}{\alpha^2} = \mathrm{R}_\alpha(0) = \mathrm{R}(r_{\max}) + \ell_+ = \max \mathrm{F} - \min \mathrm{F} + \hat{\delta}_\alpha + \ell_+,$$

with ℓ_+ our first hyperparameter *controlling how close to the origin of our spherically symmetric factor we need to be to match the maximum of Forman on the input graph.* In particular, given ℓ_+, the constraint in equation 26 yields:

$$\hat{\delta}_\alpha = \frac{2}{3\pi^2 - 2} \left(\max \mathrm{F} - \min \mathrm{F} + \ell_+ \right) + \delta$$

where $\delta > 0$ is our second hyper-parameter determining what is the range of radial coordinates needed for the curvature matching, since the smaller δ the closer to its asymptote R_α must be to take on the value $\min \mathrm{F}$. In conclusion, the choice of α is affected by two geometric hyperparameters ℓ_+, δ and is a function of the range of Forman curvatures on the given graph we want to embed in our heterogeneous model:

$$\alpha = \left(\frac{12}{\max \mathrm{F} - \min \mathrm{F} + \delta + \ell_+} \right)^{\frac{1}{2}}.$$

D FURTHER EXPERIMENTS AND DETAILS

D.1 A SYNTHETIC EXPERIMENT

Curvature is deeply related to the rate of expansion of space on manifolds, meaning that it affects whether volume of geodesic balls grow polynomially or exponentially Bishop et al. (1964). While on homogeneous manifolds the volume of a geodesic ball of given radius is independent of the position of its center, on heterogeneous manifolds the volume is *position-dependent*. To highlight this aspect in our setting, we consider a heterogeneous graph composed of a cycle and a tree (Figure 6). Note that the choice of the graph is emblematic of heterogeneous pattern since nodes inside the cycle would have constant volume growth while nodes in the tree region will have exponential volume growth. We then embed this graph in $\mathbb{H}^3 \times \mathcal{R}$ and use the normalized volume of the ambient manifold to match the volume on the graph for a given radius ρ. Explicitly, we fix a radius $\rho = 4$ and we compute the volume of each 'geodesic' ball inside a graph, i.e. $|\{x_j \in V : d_G(x_i, x_j) \leq \rho\}|$. We then use the spherical symmetry of our ambient space and the formula in equation 16 with $\rho = 4$ and y_i given by our embedded nodes. Once we normalize the volume scores on both the graph and the ambient space, we can finally compare the results as in Figure 6. We emphasize again that on a homogeneous manifold this information cannot be accessed from the actual continuum space since $|B_{g_h}(p, r)|$ is only a function of the radius but not of the base-point if (M_h, g_h) is homogeneous.

D.2 RECONSTRUCTION TASKS

We use our embeddings for the reconstruction of four real-world graphs: aves-wildbird Rossi & Ahmed (2015) (small animal network) CS-PhD De Nooy et al. (2018) (advisor-advisee relationship), web-edu Gleich et al. (2004) (web networks from the .edu domain) and Facebook Leskovec & Mcauley (2012) (dense social network).

We show that our heterogeneous embeddings perform well w.r.t. distance-based metrics (average distance distortion AD_d and mean average precision mAP) while also matching the node-wise curvature information with the pointwise scalar curvature on the manifold. To assess the quality of the latter, we introduce the *average curvature distortion*

$$\mathsf{AD}_c := \frac{1}{n} \sum_{x_i=1}^{n} \frac{|\mathrm{F}(x_i) - \mathrm{R}(f(x_i))|}{|\mathrm{F}(x_i)| + 1}.$$

We stress that the *variance* of Forman is generally high due to its dependence on the size of the degrees (Table 1). In fact, we have also confirmed experimentally that if we normalize Forman Ricci along each edge using the largest degree of the end-nodes, then AD_c is below 10^{-3} on each dataset. As baseline homogeneous embeddings, we use different products of space-forms M_h from Gu et al. (2018) and compare them to the heterogeneous embeddings constructed with the rotationally symmetric factor $M_h \times \mathcal{R}$. The results in Table 1 show that the proposed model attains reconstruction fidelity (in the sense of distance distortions) on par with the homogeneous baseline while also minimizing AD_c. In the homogeneous setting one can only match an *average* 'global curvature' as heuristically investigated in Gu et al. (2018) since the curvature is position-independent. Accordingly, computing AD_c is meaningless and we then simply report the variance of the Forman curvature as a measure of the information lost when moving from the graph curvature to the smooth manifold one.

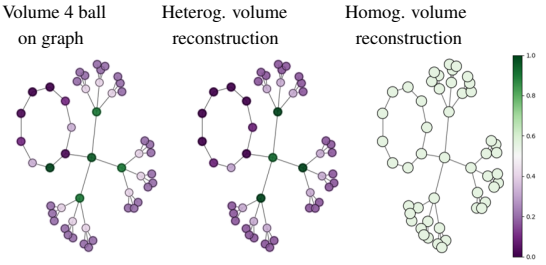


Figure 6: Volume matching.

	Aves-Wildbird	CS-PhD	WebEdu	Facebook
Improvement	49.1 %	0 %	44.2%	20.1%

Table 2: Improvement of triangle counts using curvature. Note that the CS-PhD graph only contains 4 triangles.

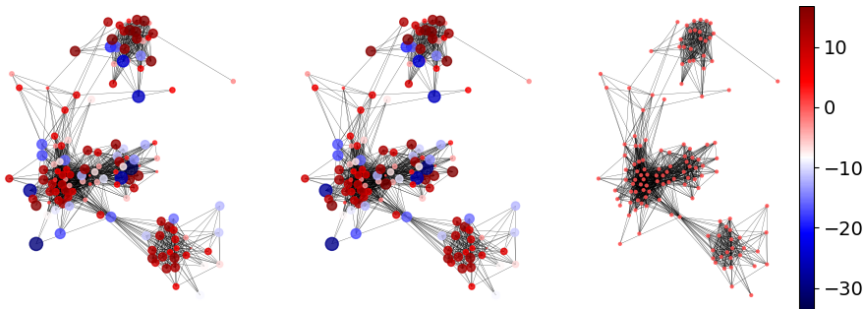


Figure 7: Curvature matching on the Aves-Wildbird dataset: true scalar Forman (left), curvature reconstructed from heterogeneous manifold (centre) and from homogeneous manifold (right).

We summarize here additional details concerning the methods and results of Table 1. We use the model proposed in Gu et al. (2018) to possibly learn optimal constant rescaling of the homogeneous factors $\mathbb{H}^5 \times \mathbb{H}^5$ and $\mathbb{H}^5 \times \mathbb{S}^5$ and we consider a training of 3000 epochs for each dataset and ambient space. Typical values of the scale parameter τ in the loss are $0.1, 0.01$, noting that this allows to minimize the curvature distortion without penalizing the distance-based one. In terms of hyperparameters ℓ_+, δ introduced in Section C.2 we usually take large values (especially for the dense network Facebook) in the range $10, 100, 1000$ which allow to avoid plateau regions of the scalar curvature profile and hence make the learning easier. This is also accounted for the initialization of the radial coordinate since once again we want to avoid flat regions of the curvature profile: since we have an explicit formula for the curvature this can be done efficiently (usually the initialization is for $r \in (0.1, 1)$).

Estimation of triangle counts. Traditional graph embedding tend to distort higher-order structures such as cycles and triangles Verbeek & Suri (2016). We verify if we can use the curvature in our heterogeneous embeddings to improve triangle counting. Given the estimated number of triangles $\sharp_\Delta(i)'$ at node i, we introduce an *average triangle distortion* AD_Δ similarly to AD_c by replacing $\mathrm{F}(x_i)$ with the actual number of triangles $\sharp_\Delta(i)$. We consider the graph nodes' embeddings in $M_h \times \mathcal{R}$ and estimate the number of triangles in two different ways: based on the *nearest-neighbour graph* and exploiting the curvature information. In the latter case, we use the formula for the Forman scalar curvature with the curvature of the manifold at respective node embedding in the place of F to estimate the number of triangles. We report the percentage improvement gained relying on curvature in Table 2.

Figure 4 shows another example of WebEdu graph reconstruction with and without the use of curvature. It highlights the advantages of heterogeneous embeddings for graph reconstruction tasks by allowing to account for the curvature.

Further details on triangle counts Given the point clouds found by the embedding into the manifold, we reconstruct the adjancency matrix as follows: we draw an edge between a pair of nodes (i, j) if the distance between the corresponding embedded points y_i, y_j is lower than a certain threshold ρ, i.e. if $d_M(y_i, y_j) \leq \rho$. Self-loops are then removed. The threshold ρ is tuned on a validation set that is built drawing randomly 10% of the nodes of the dataset. The tuning aims at minimizing the reconstruction error between the reconstructed and real graph: given A the adjacency matrix of the graph and A_ρ the adjacency matrix associated with the ρ-nearest neighbour graph, we tune ρ to minimize $\|A - A_\rho\|$ on the validation set. More advanced ways for graph reconstruction and link predictions exist in the literature (see for example Nickel & Kiela (2017)).

Triangle distortion				
$\lvert\Delta\rvert$, avg Δ_i	**Aves-Wildbird** 9270, 70.76	**CS-PhD** 4, 0.004	**WebEdu** 10058, 3.31	**Facebook** 1612010, 399.2
$\mathbb{H}^5 \times \mathbb{H}^5$	0.212	0.004	0.658	0.518
$\mathbb{H}^5 \times \mathbb{H}^5 \times \mathcal{R}$	0.108	0.004	0.369	0.414

Table 3: Performance on the triangle counting task.

Given our best nearest neighbour reconstruction adjacency A_ρ and our manifold curvature values $\mathrm{R}(y_i)$ we reconstruct the number of triangles using equation 1 and its degree average where the true adjacency is replaced by the reconstructed one. Explicitly:

$$(3 \cdot 2)\gamma \sharp_\Delta(i) = d_\rho(i)\mathrm{R}(y_i) - \sum_j (A_\rho)_{ij}(4 - d_\rho(i) - d_\rho(j)),$$

where the extra 2 factor on the LHS derives from counting each triangle twice in the formula $2\sharp_\Delta(i) = \sum_{j\sim i} \sharp_\Delta(i,j)$. For the results reported in Table 3 we take $\gamma = 4$ in the weighting of triangles. We also note that the 0% improvement over the CS-PhD dataset is to ascribe to the extremely low density of the true graph (with only 4 triangles overall): both methods reach low average distortion - albeit in this case not highly indicative.

Curvature correction. Once the reconstructed adjacency A_ρ (and hence a reconstructed graph G_ρ) is available, one can compute the node-wise Forman curvature, $\mathrm{F}_\rho(i)$ with $i \in V$. Since in our embeddings the curvature on the manifold R is a good proxy of the curvature of the graph, one can use the discrepancy $|\mathrm{R}(y_i) - \mathrm{F}_\rho(i)|$ to identify the points where the reconstruction is poor. Indeed, the discrepancy $|\mathrm{R}(y_i) - \mathrm{F}_\rho(i)|$ translate the quality of the reconstruction of the 2-hop neighborhood of the node i, by definition of Forman curvature. *How to best exploit this additional information in reconstruction tasks and link prediction is of interest on its own and goes beyond the scope of the work*. However, we conducted preliminary experiments resorting on a simple curvature correction that acts as follows:

- Compute $err_i = |\mathrm{R}(y_i) - \mathrm{F}_\rho(i)|$ for each node i
- Identify the nodes where the error err_i is bigger, e.g. the nodes where err_i is above the 90% percentile.
- For these nodes, increase / decrease the distance threshold that governs the edge, obtaining a new graph G'_ρ that is locally modified from the reconstructed graph G_ρ.
- Compute the curvature of nodes of the new graph G'_ρ and compare it with the curvature of the corresponding points on the manifolds. If the discrepancy between the curvature decreases, accept the change. Otherwise reject it.

We have tested this method on WebEdu attaining a 22.6% improvement in the reconstruction (see Figure 4).

D.3 Generating graphs from heterogeneous manifolds

Inspired by Cruceru et al. (2021), we study the advantage of heterogeneous manifolds from the perspective of random graphs. The focus is on how one can use the curvature information to generate graphs that are highly *heterogeneous* and exhibit localized dense community structure while preserving scale-free properties common to *small world networks* Watts & Strogatz (1998); Newman (2000; 2003). We consider the following setting: we generate graphs of 500 nodes from $\mathbb{H}^3$ and $\mathbb{H}^3 \times \mathcal{R}$ using uniform sampling on the tangent space. We then test two approaches to promote formation of community structures, measured by the size of maximal cliques. On $\mathbb{H}^3$, since the only geometric quantity we can vary is the distance, we sample nearest-neighbour graphs with increasing distance threshold. On $\mathbb{H}^3 \times \mathcal{R}$, we instead combine distance and curvature: we increase the distance threshold only for nodes sampled from more positively-curved regions (see below for details). Figure 8 (left-to-right) depicts the degree distributions (averaged over 100 runs using Wasserstein barycenters Agueh & Carlier (2011)) for three cases: sampling from $\mathbb{H}^3$ with unit distance threshold (no dense community structure), with larger distance threshold to encourage clique formation, and

relying on curvature in $\mathbb{H}^3 \times \mathcal{R}$ to attain similar clique sizes. We see how the curvature gives rise to heterogeneous density on the graph hence achieving dense community structure while preserving the scale-free profile. On the other hand, sampling from a homogeneous manifold with different thresholds cannot attain dense cliques without losing the power-law degree distribution due to a homogeneous increase in density.

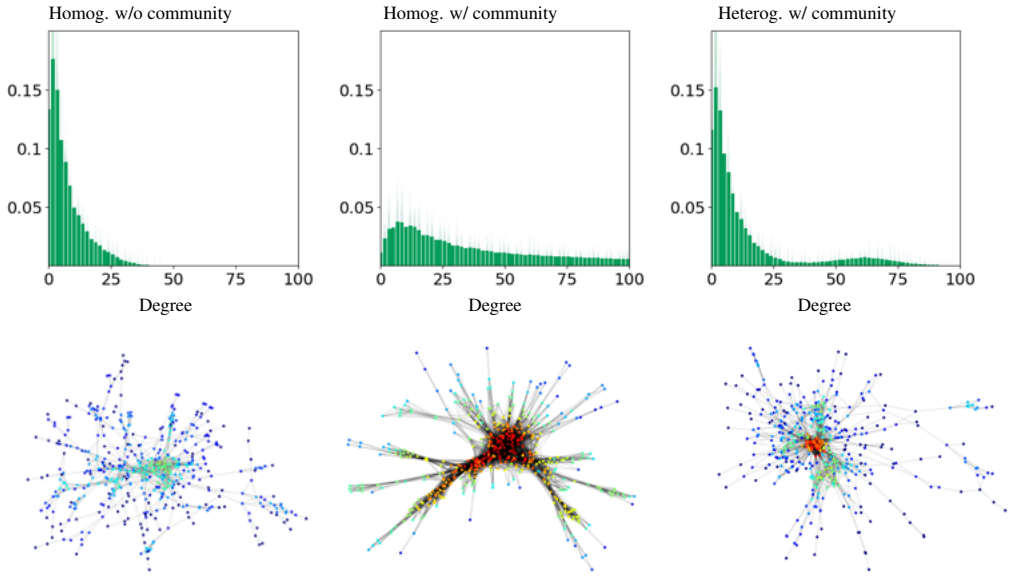


Figure 8: Degree distribution and generated examples of graphs sampled from hyperbolic space (left), after the creation of community using distance (centre) and curvature (right). Nodes are colored by the number of triangles (blue corresponding to lower values, red to bigger values).

Manifold random graphs Here we comment more on our random graph generation. We consider a three dimensional hyperbolic space $\mathbb{H}^3$ and we follow the sampling procedure adopted for example in Cruceru et al. (2021): namely, one samples uniformly tangent vectors at some fixed point (this is not important due to homogeneity) and then use the exponential map to generate points inside the manifold. We observe that this approach is biased since it does not account for the underlying geometry (i.e. the Riemannian measure) but only sees the 'flat' geometry of the tangent spaces. Nonetheless, for our purposes of random generation we prefer to stick to this easier uniform tangent sampling. In fact, if we actually encoded the hyperbolic Riemannian measure, then the sampling would have an even more manifest scale-free profile (since points on the Poincaré disk would be on average closer to the boundary) as shown in Cruceru et al. (2021).

Once we have a point cloud inside the Cartesian product of the Poincaré disk (and our spherically symmetric extension $\mathbb{H}^3 \times \mathcal{R}$), we construct the *nearest neighbour graph* G with adjacency A using a **distance threshold** ρ, meaning that $A_{ij} = 1$ if points y_i and y_j are at geodesic distance smaller or equal than ρ. In general, graphs uniformly sampled from a hyperbolic geometry without accounting for heterogeneous curvature will exhibit small-world network features as power-law degree distribution Krioukov et al. (2010), however they will generally lack community structure (cliques). We then set the following:

Goal: *Sample random graphs of 500 nodes that have one large community (as measured by the existence of a clique of* $\sim 45 - 50$ *nodes) while preserving the scale-free behaviour of the density (degree).*

For the statistics reported below we sample 100 random graphs for each given threshold.

Approach one: Increase the distance threshold ρ In one case, where we simply sample points from the hyperbolic space, we consider increasing thresholds ρ to improve the average density.

While this allows for formation of dense community structures and achieves higher mean clustering, the higher density is distributed uniformly on the graph due to the homogeneity of the continuous manifold. In fact, the variance of the degree increases dramatically too, ultimately resulting in a failure to preserve the scale-free property when arriving to large cliques. On the other hand, the variance of the clustering coefficient decreases by more than 50 $\%$, highlighting how now in most of the graph the probability of triangle formation is uniformly high. This is all summed up in the statistics reported in Table 4.

Homogeneous	$\rho = 1$	$\rho = 1.2$	$\rho = 1.5$	$\rho = 1.7$	$\rho = 2$
variance degree	6.79 ± 1.85	10.72 ± 3.05	18.27 ± 3.99	23.71 ± 5.76	40.36 ± 8.95
mean degree	7.33 ± 1.47	11.34 ± 2.73	19.33 ± 3.70	25.94 ± 5.64	47.24 ± 10.23
variance clustering	0.29 ± 0.01	0.27 ± 0.01	0.22 ± 0.01	0.19 ± 0.01	0.15 ± 0.01
mean clustering	0.42 ± 0.02	0.53 ± 0.02	0.63 ± 0.01	0.67 ± 0.01	0.73 ± 0.01
size largest clique	10.6 ± 1.97	15.28 ± 3.18	23.84 ± 4.58	29.22 ± 6.07	49.96 ± 9.39

Table 4: Statistics of the random graphs sampled from the homogeneous model $\mathbb{H}^3$ for different *distance thresholds*. The formation of dense community structure can only occur uniformly at the cost of the scale-free property of the networks.

Approach two: increase the connectivity among positively curved points For the point cloud sampled uniformly[3] in $\mathbb{H}^3 \times \mathcal{R}$ we can also leverage the curvature information, meaning that now differently from the homogeneous space to each point y_i we can also associate position-dependent curvature information $\mathrm{R}_\alpha(y_i)$. In particular we fix α in equation 15 and some curvature threshold ℓ and assign a connection between any pair of points sampled from $\mathbb{H}^3 \times \mathcal{R}$ with both curvatures larger than ℓ and within a distance threshold ρ we now vary again as above. We report the results in Table 5: we point out that now we reach a large dense clique (community) structure while still preserving the scale-free profile as shown in the degree distribution in Figure 8, the mean degree and its variance. The degree distribution is representative of the 100 runs, as it is obtained computing the average (or barycenter) of the degree distributions of all runs using Wasserstein distance. Wasserstein distance is sensitive to the shape and geometry of the probability distributions and therefore particularly suitable to compute the average of histograms, preserving their shape (Agueh & Carlier (2011); Cuturi & Doucet (2014); Luise et al. (2019)). Moreover, we observe how the variance of the clustering coefficient has not been affected significantly meaning that our sampling has managed to give rise to a highly heterogeneous density distribution. This is just a simple application of how heterogeneous manifolds could potentially be used to generate believable graphs that share many properties with real ones.

Heterogeneous	$\rho = 1$	$\rho = 2.5$	$\rho = 4$	$\rho = 5.5$	$\rho = 7.0$
variance degree	6.34 ± 1.7	8.26 ± 2.8	12.9 ± 3.2	16.21 ± 2.7	19.4 ± 3.51
mean degree	7.09 ± 1.37	8.46 ± 2.20	10.99 ± 2.58	12.33 ± 1.81	14.06 ± 2.47
variance clustering	0.28 ± 0.01	0.28 ± 0.01	0.28 ± 0.01	0.29 ± 0.01	0.30 ± 0.01
mean clustering	0.42 ± 0.02	0.43 ± 0.02	0.45 ± 0.02	0.46 ± 0.02	0.47 ± 0.02
size largest clique	10.6 ± 2.29	11.94 ± 3.17	24.9 ± 6.31	35.9 ± 5.72	47.2 ± 8.24

Table 5: Statistics of the random graphs sampled from our heterogeneous model $\mathbb{H}^3 \times \mathcal{R}$ for different *curvature thresholds*. We note how compared to the case where sampled points all come with same curvature from the ambient space, in this case we can attain large clique sizes without affecting significantly the variance of the clustering and of the degree distribution.

Limitations and future directions. We have restricted our discussion to embedding of undirected graphs. Embeddings of directed graphs into (pseudo)-Riemannian manifolds have recently been studied by Sim et al. (2021). In future work, we will study extensions of the proposed framework to such settings. Second, we used a fixed rotationally symmetric function φ which determines the curvature profile of the ambient space. It is possible to make it *learnable* in future extensions.

[3]We point out here that the sampling occurs in a compact region. In the case of $\mathcal{R}$ we consider radial coordinates sampled in the interval $(0, 2)$.

Third, exploiting Ricci curvature information along walks, more sophisticated anisotropic curvature matching models could be investigated to go beyond the scalar distribution at nodes. Finally, the family of manifolds we consider is a subclass of a more general ensemble of *warped products* that will be investigated in the future. In particular, one can also explore using multiple rotationally symmetric copies to better account for tensorial curvature information.