Investigating Zero-Shot Size Transfer of Graph Neural Differential Equations for Learning Graph Diffusion Dynamics

Charles Kulick Björn Birnir Sui Tang
Department of Mathematics
University of California, Santa Barbara
Santa Barbara, CA 93106
{charles.kulick, bjorn.birnir, suitang}@ucsb.edu

Abstract

Graph Neural Differential Equations (GNDEs) provide a powerful framework for learning continuous-time dynamics on discrete graphs. A key advantage of GNDEs is their potential to transfer learned parameters from small training graphs to much larger test graphs. While recent theory establishes convergence guarantees for this size transfer, its practical behavior in trained models remains unclear. In this work, we empirically investigate the zero-shot size transferability of GNDEs for learning graph diffusion dynamics that admit a graphon limit, and reveal how transfer accuracy depends on the underlying dynamics and graph structure. In particular, we observe a clear convergence rate when fitting GNDEs to linear diffusion dynamics across graph sizes, a phenomenon absent in nonlinear settings, raising new theoretical questions about the mechanisms underlying transfer learning.

1 Introduction

Modeling and learning continuous dynamical processes on large networks remains a central challenge in scientific machine learning. *Graph Neural Differential Equations* (GNDEs) [Poli et al., 2019, Liu et al., 2025] provide a principled framework for representing such dynamics in continuous time. By combining the structural awareness of Graph Neural Networks (GNNs) [Scarselli et al., 2008] with the flexibility of Neural Ordinary Differential Equations (Neural ODEs) [Chen et al., 2018], GNDEs learn vector fields that govern the temporal evolution of signals on graphs, enabling data-driven modeling of diffusion, reaction, and consensus phenomena [Han et al., 2022, Huang et al., 2023, 2024, Luo et al., 2023].

However, training GNDEs on large graphs is computationally expensive [Finzi et al., 2023, Liu et al., 2025]. A promising alternative is to utilize their potential *size transferability* [Ruiz et al., 2020, Levie et al., 2021], wherein a GNDE trained on small graphs may generalize, without retraining, to much larger graphs generated from the same structural rule. This is particularly relevant when the underlying physical dynamics, such as heat diffusion, Allen–Cahn, or consensus dynamics, possess well-defined graphon limits [Medvedev, 2013]. These limits capture continuum behaviors that remain consistent across scales, mirroring many physical systems which exhibit self-similar interaction laws. Understanding whether GNDEs can exploit this inherent regularity to generalize across graph sizes makes zero-shot transfer both theoretically grounded and practically appealing.

Building on recent theoretical results demonstrating the size transferability of GNDEs [Yan et al., 2025] which shows their solutions converge on graphs sampled from graphons in the same manner as graph diffusion dynamics (see Section C), we empirically investigate whether this convergence translates into scalable generalization. Specifically, we train GNDEs to learn graph diffusion dynamics

1st Conference on Topology, Algebra, and Geometry in Data Science (TAG-DS 2025).

on small deterministic graphs sampled from a fixed graphon and evaluate their predictive performance on increasingly larger graphs generated from the same graphon, without any fine-tuning or retraining.

This work makes the following contributions:

- We quantify the scaling of GNDE predictive accuracy with graph size, identifying the graph size required to maintain performance and revealing how this threshold depends on both the underlying dynamics and the geometric complexity of the graphon.
- For linear diffusion dynamics, we fit GNDEs on graphs of different sizes and use the learned hyperparameters to predict the corresponding graphon-limit solution. The resulting prediction errors exhibit a clear rate of convergence toward the graphon solution, a phenomenon absent in the nonlinear cases which raises a new direction for analyzing transfer learning of linear dynamics in continuous-depth settings.

2 Notation and preliminaries

Let $\mathbb{N} := \{1, 2, \ldots\}$. For $n \in \mathbb{N}$, let $[n] := \{1, 2, \ldots, n\}$. We denote the unit interval as $I := [0, 1]$ and $I^2 := I \times I$. For an interval $J \subseteq I$, by $|J|$ we denote the length of J, and we define the indicator function $\chi_J : J \to \{0, 1\}$ as $\chi_J(u) := 1$ if $u \in J$ and $\chi_J(u) := 0$ otherwise.

Graphons and deterministic graphon sampling Throughout this work, we denote discrete graphs by $\mathcal{G}_n$ with weighted adjacency matrices $\mathbf{W}_n \in [0, 1]^{n \times n}$ and one-dimensional node features $\mathbf{Z}_n \in \mathbb{R}^n$. A *graphon* is a symmetric measurable function $\mathbf{W} : I^2 \to [0, 1]$ that describes pairwise interaction strengths between any two points $u, v \in I$ and serves as the continuous limit of dense graphs. We associate to $\mathbf{W}$ a *graphon feature function* $\mathbf{Z} \in L^\infty(I)$, where $\mathbf{Z}(u)$ represents the node feature at position u. Given a fixed graphon $\mathbf{W}$, we construct a deterministic graph $\mathcal{G}_n$ by placing nodes at $u_i = (i - 1)/n$ for $i \in [n]$ and setting the edge weights $[\mathbf{W}_n]_{ij} = \mathbf{W}(u_i, u_j)$, and node feature function $[\mathbf{Z}_n]_i = \mathbf{Z}(u_i)$. To link discrete and continuous representations, we partition I into intervals $I_j = [\frac{j-1}{n}, \frac{j}{n})$ and define the **induced graphon and feature functions**,

$$\overline{\mathbf{W}}_n(u, v) = \sum_{i,j=1}^{n} [\mathbf{W}_n]_{ij} \chi_{I_i}(u) \chi_{I_j}(v), \qquad \overline{\mathbf{Z}}_n(u) = \sum_{i=1}^{n} [\mathbf{Z}_n]_i \chi_{I_i}(u). \tag{1}$$

This deterministic construction, following Medvedev [2013], Ruiz et al. [2021], provides a consistent correspondence between discrete graphs and their graphon limits and a principled way to compare the graph feature functions of varying size. More details are presented in Section A.1.

Graph Neural Differential Equations A GNDE is a Neural ODE with the right hand side function parameterized by a GNN architecture Φ_n:

$$\begin{aligned} \frac{d}{dt} \mathbf{X}_n(t) &= \Phi_n(\mathbf{S}_n; \mathbf{X}_n(t); \mathbf{\Theta}(t)) \\ \mathbf{X}_n(0) &= \mathbf{Z}_n \in \mathbb{R}^n \end{aligned} \tag{2}$$

where $\mathbf{S}_n$ is a descriptor matrix of graph topology often taken as a scaled version of the weighted adjacency matrix; $\mathbf{X}_n(t) \in \mathbb{R}^n$ denotes the node features at time t; $\mathbf{\Theta}(t)$ are learnable time-varying parameters; and $\mathbf{Z}_n$ is the initial condition. Such GNDEs are evolved on a time interval $[0, T]$ to model dynamics on graphs. In our experiments, we consider the continuous-depth spectral GNN models Poli et al. [2019] with full details in Appendix B. It is shown in Yan et al. [2025] that (2) converges to a graphon limit when the underlying graphs converge to a graphon.

3 Numerical results

Diffusion dynamics on graphs sampled from graphons We choose 4 representative diffusion dynamics (see Table 1) defined on graphs sampled from 5 different graphons: three binary graphons with increasing boundary complexity and two smooth graphons (Figure 1). For each graphon, we construct discrete graphs with sizes n ranging from 25 to 300 as outlined in Appendix A.2.

Table 1: Graph-based dynamical systems used as teacher models, where w_{ij} denotes the adjacency weights and $[\mathbf{X}_n]_i$ represents the node feature at vertex i. Their graphon limits and convergence rates are discussed in Section C.

Dynamics	**Equation on node i**	**Description**
Linear Heat	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_{j=1}^{n} w_{ij}\left([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\right)$	Classical diffusion (graph Laplacian flow); models linear heat propagation on graphs.
Nonlinear Heat	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_{j=1}^{n} w_{ij}\, \sin\left([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\right)$	Diffusion with smooth nonlinear coupling; models energy-dissipating heat flow on graphs.
Allen–Cahn	$\frac{d}{dt}[\mathbf{X}_n]_i = -\frac{\epsilon^2}{n}\sum_{j=1}^{n} w_{ij}\left([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\right) + [\mathbf{X}_n]_i - [\mathbf{X}_n]_i^3$	Double-well potential dynamics; describes phase separation and bistability.
Nonlinear Consensus	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_{j=1}^{n} w_{ij}\, \frac{[\mathbf{X}_n]_j - [\mathbf{X}_n]_i}{1 + ([\mathbf{X}_n]_j - [\mathbf{X}_n]_i)^2}$	Nonlinear averaging with saturating influence; models bounded-confidence consensus.

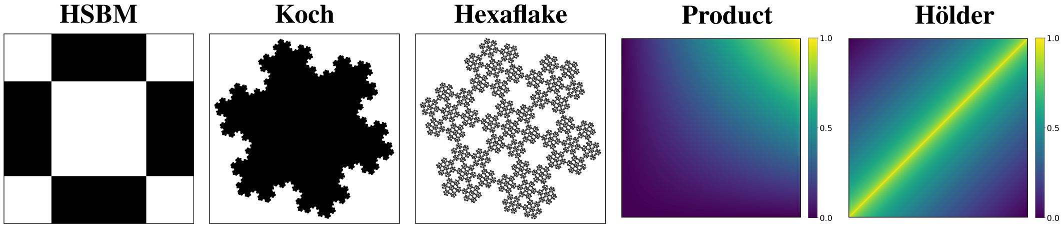

Figure 1: The five graphons used in our experiments (left to right): three binary graphons HSBM, Koch, and Hexaflake with increasing boundary complexity; and two smooth weighted graphons Product (Lipschitz) and Hölder ($\frac{1}{2}$-Hölder). **The theoretical convergence-rate exponents of graph diffusion dynamics in Table 1 to their graphon counterpart are: HSBM -0.50, Koch -0.37, Hexaflake -0.115, Product -1, and Hölder -0.5.**

Training setup On each discrete graph, we evolve the four dynamics of Table 1 on 25 random Fourier initial conditions and save initial condition and final state pairs as training data. We build GNDE models utilizing the code of Poli et al. [2019]. All GNDE models, regardless of graph size, are parameterized by the spectral GCN models of Appendix B with $K = 2$ and have an input and output layer with dimension 1 and two hidden layers with an ambient dimension of 16. This model architecture was primarily selected by grid search (Section D.1) with the choice of $L = 2$ hidden layers informed by preliminary testing and chosen to balance final performance and training cost. Models were trained with MSE loss on final state prediction of the dynamics and optimized with Adam [Kingma and Ba, 2014] for 200 epochs, which was observed as sufficient for convergence for the largest graphs. We utilized only standard methods for training as our goal was not state of the art performance but rather realistic trained GNDE models, and our code is available on GitHub[1]. Computational complexity analysis, additional plots, and implementation details are provided in Section D.1.

Evaluation To evaluate the size transfer performance of trained models, we construct a larger graph with 1000 nodes. We transfer weights from each model trained on smaller graphs of size n to this larger graph and predict using test set initial conditions. We measure and report the relative L^2 error of the predicted final state at time T on the larger graph. Mean and standard deviation of this relative error is reported across ten trials for each graph size and graphon. The mean relative errors are plotted in Figure 2 for the linear heat dynamics, and reported in Table 2 for the remaining dynamics. Results with error bars are reported in Appendix Figure 3.

Analysis Across dynamics, GNDEs achieve low error even at small training sizes n. For smoother graphons or those with simpler boundaries, modest increases in n yield additional accuracy gains, consistent with faster convergence to the graphon limit, whereas graphons with rough or complex

[1] `https://github.com/CharlesKulick/GNDE_Size_Transfer`

Table 2: Relative L^2 error (mean ± std) averaged over 10 runs across various dynamics with results displayed for each graphon and discrete training graph size n.

Dynamics	n	**Product**	**Hölder**	**HSBM**	**Koch**	**Hexaflake**
Nonlinear Heat	25	0.045 ± 0.003	0.074 ± 0.005	0.202 ± 0.103	0.035 ± 0.003	0.054 ± 0.014
	50	0.045 ± 0.003	0.071 ± 0.006	0.121 ± 0.015	0.029 ± 0.007	0.044 ± 0.012
	75	0.046 ± 0.003	0.072 ± 0.006	0.110 ± 0.031	0.028 ± 0.008	0.041 ± 0.011
	100	0.046 ± 0.003	0.071 ± 0.006	0.096 ± 0.024	0.030 ± 0.006	0.042 ± 0.011
	200	0.045 ± 0.003	0.072 ± 0.007	0.084 ± 0.009	0.029 ± 0.006	0.039 ± 0.011
	300	0.045 ± 0.003	0.072 ± 0.007	0.080 ± 0.008	0.030 ± 0.006	0.039 ± 0.010
Nonlinear Consensus	25	0.201 ± 0.075	0.278 ± 0.015	0.269 ± 0.015	0.070 ± 0.010	0.136 ± 0.031
	50	0.119 ± 0.017	0.239 ± 0.026	0.263 ± 0.014	0.062 ± 0.012	0.142 ± 0.046
	75	0.110 ± 0.010	0.235 ± 0.020	0.263 ± 0.011	0.059 ± 0.006	0.122 ± 0.036
	100	0.106 ± 0.011	0.233 ± 0.022	0.267 ± 0.014	0.060 ± 0.005	0.124 ± 0.049
Allen-Cahn	25	0.129 ± 0.181	0.034 ± 0.003	0.037 ± 0.003	0.131 ± 0.186	0.131 ± 0.179
	50	0.126 ± 0.186	0.033 ± 0.002	0.032 ± 0.001	0.127 ± 0.188	0.035 ± 0.005
	75	0.033 ± 0.002	0.032 ± 0.002	0.031 ± 0.002	0.125 ± 0.187	0.032 ± 0.002
	100	0.031 ± 0.001	0.030 ± 0.002	0.031 ± 0.001	0.125 ± 0.187	0.033 ± 0.003

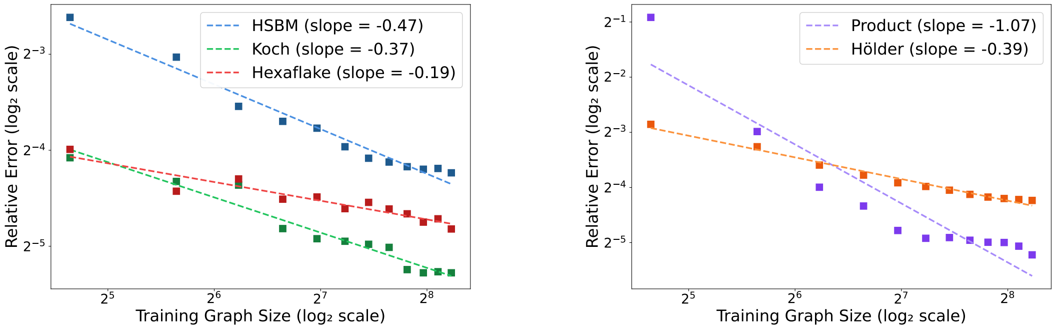

Figure 2: Mean relative L^2 error for the **linear heat equation** across ten trials, plotted against training graph size on a log-log plot to provide rate of convergence estimates. Left: Unweighted graphons (HSBM, Koch, and Hexaflake). Right: Weighted graphons (Product, Hölder). The observed rates match the rates of graph solutions to graphon counterparts as reported in caption of Figure 1.

boundaries typically require substantially larger n to improve. This contrast is most evident when comparing diffusion-like dynamics (nonlinear heat) to harder, weakly coupled regimes (nonlinear consensus).

Linear diffusion case For the linear heat equation, the decay of transfer error with n closely matches the predicted graphon-limit rates (see Figure 2 and Table 3): the observed exponents align with theory across all five graphons, substantiating the dependence of convergence on boundary complexity for binary graphons and on Hölder regularity for weighted graphons. Error bars are provided in Figure 3 in Section D.

Table 3: Theoretical and observed exponents for the rate of convergence for the linear heat equation across five tested graphons. Note the close agreement between the slopes in Figure 2 and the theoretical rates.

Graphon	**Theoretical Exponent**	**Observed Exponent**
HSBM	-0.5	-0.47
Koch	-0.37	-0.37
Hexaflake	-0.115	-0.19
Product	-1	-1.07
Hölder	-0.5	-0.39

Beyond linear diffusion For nonlinear settings, errors tend to plateau once $n \approx 100$, indicating that small training graphs already suffice to transfer accurately to the larger graph with 1000 nodes. In Figure 4, we provide visual evidence for the close match between transfer predictions and true dynamics, showing the high accuracy of the learned model. The absence of clear rate decay in these regimes is likely due to increased optimization difficulty and weaker effective coupling (e.g., saturation in consensus), rather than a fundamental failure of size transfer; see Figure 5 for representative loss curves for some evidence of the qualitative difference in linear and nonlinear dynamics training.

We also investigated more expressive models through a small grid search to ensure the observed plateau is not solely due to model capacity limitations. Using models with $L \in \{2, 3, 4\}$ hidden layers, hidden dimensions of $H \in \{16, 32, 64\}$, and K-hop neighborhoods with $K \in \{2, 3, 4\}$, we verified similar behaviors in nonlinear heat dynamics. The difference in mean relative transfer error between our chosen architecture and the best model from this grid is a 20% relative reduction at $n = 100$, which is a nontrivial relative difference but a small absolute difference. This behavior provides initial evidence that more expressive model architectures may not fully alleviate the observed phenomenon, but a more thorough architecture search is required to fully conclude the role of longer-range message passing and model expressivity. We leave this investigation for future work.

4 Limitations and future work

Our experiments are restricted to moderate graph sizes due to the computational cost of training GNDEs. As shown in Appendix D, the variance across initial conditions remains noticeable, reflecting the difficulty of optimizing continuous-depth graph models. Improved solvers or variance-reduction strategies could help enhance stability and scalability. Future work will explore extending GNDEs beyond message passing toward architectures with richer global context. Graph transformers, in particular, offer a promising direction for modeling long-range dependencies and heterogeneous interactions. Integrating transformer components into continuous-depth formulations and developing *foundation models* for graph dynamics, pretrained across diverse graph families and transferable to new regimes, represent exciting next steps toward scalable scientific machine learning on graphs.

References

Christian Borgs, Jennifer Chayes, László Lovász, Vera Sós, and Katalin Vesztergombi. Limits of randomly grown graph sequences. *European Journal of Combinatorics*, 32(7):985–999, 2011. ISSN 0195-6698. doi: https://doi.org/10.1016/j.ejc.2011.03.015. URL https://www.sciencedirect.com/science/article/pii/S0195669811000667. Homomorphisms and Limits.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs, 2014. URL https://arxiv.org/abs/1312.6203.

John C. Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, 2008. doi: 10.1002/9780470753767.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 3844–3852, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Marc Finzi, Andres Potapczynski, Matthew Choptuik, and Andrew Gordon Wilson. A stable and scalable method for solving initial value pdes with neural networks. *arXiv preprint arXiv:2304.14994*, 2023.

Zhichao Han, David S Kammer, and Olga Fink. Learning physics-consistent particle interactions. *PNAS nexus*, 1(5):pgac264, 2022.

Zijie Huang, Yizhou Sun, and Wei Wang. Generalizing graph ode for learning complex system dynamics across environments. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 798–809. Association for Computing Machinery, 2023.

Zijie Huang, Jeehyun Hwang, Junkai Zhang, Jinwoo Baik, Weitong Zhang, Dominik Wodarz, Yizhou Sun, Quanquan Gu, and Wei Wang. Causal graph ode: Continuous treatment effect modeling in multi-agent dynamical systems. In *Proceedings of the ACM on Web Conference 2024*, pages 4607–4617, 2024.

John E. Hutchinson. Fractals and self similarity. *Indiana University Mathematics Journal*, 30(5): 713–747, 1981. ISSN 00222518, 19435258. URL http://www.jstor.org/stable/24893080.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv e-prints*, pages arXiv–1412, 2014.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Ron Levie, Wei Huang, Lorenzo Bucci, Michael Bronstein, and Gitta Kutyniok. Transferability of spectral graph convolutional neural networks. *Journal of Machine Learning Research*, 22(272): 1–59, 2021.

Zewen Liu, Xiaoda Wang, Bohan Wang, Zijie Huang, Carl Yang, and Wei Jin. Graph odes and beyond: A comprehensive survey on integrating differential equations with graph neural networks. *arXiv preprint arXiv:2503.23167*, 2025.

László Lovász. *Large networks and graph limits*, volume 60. American Mathematical Soc., 2012.

Xiao Luo, Jingyang Yuan, Zijie Huang, Huiyu Jiang, Yifang Qin, Wei Ju, Ming Zhang, and Yizhou Sun. Hope: High-order graph ode for modeling interacting dynamics. In *International Conference on Machine Learning*, pages 23124–23139. PMLR, 2023.

Georgi S. Medvedev. The nonlinear heat equation on dense graphs and graph limits, 2013. URL https://arxiv.org/abs/1302.5804.

Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs, 2016. URL https://arxiv.org/abs/1605.05273.

Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106 (5):808–828, 2018.

Michael Poli, Stefano Massaroli, Junyoung Park, Atsushi Yamashita, Hajime Asama, and Jinkyoo Park. Graph neural ordinary differential equations. *arXiv preprint arXiv:1911.07532*, 2019.

Luana Ruiz, Luiz Chamon, and Alejandro Ribeiro. Graphon neural networks and the transferability of graph neural networks. *Advances in Neural Information Processing Systems*, 33:1702–1712, 2020.

Luana Ruiz, Luiz FO Chamon, and Alejandro Ribeiro. Graphon signal processing. *IEEE Transactions on Signal Processing*, 69:4961–4976, 2021.

Carl Runge. Über die numerische auflösung von differentialgleichungen. *Mathematische Annalen*, 46(2):167–178, 1895.

Aliaksei Sandryhaila and José MF Moura. Discrete signal processing on graphs. *IEEE transactions on signal processing*, 61(7):1644–1656, 2013.

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.

David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

Mingsong Yan, Charles Kulick, and Sui Tang. Graph neural differential equations in the infinite-node limit: Convergence and rates via graphon theory. In *New Perspectives in Graph Machine Learning*, 2025.

A Graphon theory

Graphs and graphons A **graph** is defined as a triple $\mathcal{G}_n = \langle V(\mathcal{G}_n), E(\mathcal{G}_n), \mathbf{W}_n \rangle$ where $V(\mathcal{G}_n) = [n]$ is the vertex set, $E(\mathcal{G}_n) \subseteq V(\mathcal{G}_n) \times V(\mathcal{G}_n)$ is the edge set, and $\mathbf{W}_n \in [0,1]^{n \times n}$ is the weighted adjacency matrix with entries $[\mathbf{W}_n]_{ij} = w_{ij}$. Here, $w_{ij} = 0$ if and only if $(i,j) \notin E(\mathcal{G}_n)$. A graph is unweighted if $\mathbf{W}_n \in \{0,1\}^{n \times n}$. We consider *simple* graphs $\mathcal{G}_n$, which are unweighted graphs containing no self-loops with $\mathbf{W}_n$ symmetric. For our focus on heat dynamics, we consider graphs $\mathcal{G}_n$ with one-dimensional node features, denoted by the vector $\mathbf{Z}_n \in \mathbb{R}^n$.

A **graphon** is a bounded, symmetric, measurable function $\mathbf{W} : I^2 \to I$. Graphons are a continuous generalization of a discrete graph, with a continuum of nodes $x \in I$ and weights given by function evaluation $\mathbf{W}(x,y)$ for nodes $x, y \in I$. In a precise sense, graphons can represent the limiting object of a sequence of discrete graphs. Similar to graphs, an unweighted graphon takes only binary values, $\mathbf{W} : I^2 \to \{0,1\}$. We also define the graphon feature function $\mathbf{Z} \in L^\infty(I)$ over the graphon $\mathbf{W}$ where $\mathbf{Z}(u)$ is the node feature for every $u \in I$.

A.1 Graphon convergence theory

We briefly review graphons as limits of graph sequences, following Borgs et al. [2011], Lovász [2012].

Homomorphism density Let $\mathcal{G}_n = \langle V_n, E_n, \mathbf{W}_n \rangle$ be a weighted undirected graph with $|V_n| = n$ and $\mathbf{W}_n \in [0,1]^{n \times n}$. For a finite simple graph (motif) $\mathcal{F} = (V(\mathcal{F}), E(\mathcal{F}))$, a homomorphism $\phi : \mathcal{F} \to \mathcal{G}_n$ is adjacency-preserving. The (weighted) homomorphism density is

$$t(\mathcal{F}, \mathcal{G}_n) = \frac{\sum_\phi \prod_{(i,j) \in E(\mathcal{F})} [\mathbf{W}_n]_{\phi(i)\phi(j)}}{n^{|V(\mathcal{F})|}}. \tag{3}$$

Let $\mathbf{W} : I^2 \to I$ be symmetric and measurable. Then

$$t(\mathcal{F}, \mathbf{W}) = \int_{I^{|V(\mathcal{F})|}} \prod_{(i,j) \in E(\mathcal{F})} \mathbf{W}(x_i, x_j) \prod_{i \in V(\mathcal{F})} dx_i. \tag{4}$$

Convergence of graph sequences A sequence $\{\mathcal{G}_n\}$ *converges* if $t(\mathcal{F}, \mathcal{G}_n)$ converges for every finite motif $\mathcal{F}$. In that case there exists a graphon $\mathbf{W}$ such that $\lim_{n \to \infty} t(\mathcal{F}, \mathcal{G}_n) = t(\mathcal{F}, \mathbf{W})$ for all $\mathcal{F}$, and every graphon arises as such a limit [Lovász, 2012].

Relabelings and permutation–invariant convergence Graphs and graphons are defined up to node relabelings. For finite graphs, a permutation $\pi \in S_n$ acts via the permutation matrix Π_π, producing the relabeled graph $\pi(\mathcal{G}_n)$ with adjacency $\Pi_\pi^\top \mathbf{W}_n \Pi_\pi$ and features $\Pi_\pi \mathbf{Z}_n$. On the continuum, a measure-preserving bijection (mpb) $\phi : I \to I$ acts on a graphon and a feature function by $\mathbf{W}^\phi(u,v) = \mathbf{W}(\phi(u), \phi(v))$ and $\mathbf{Z}^\phi(u) = \mathbf{Z}(\phi(u))$. Define the cut norm $\|U\|_\square = \sup_{S,T \subset I} \left| \int_{S \times T} U(u,v)\, du\, dv \right|$ and the cut distance

$$\delta_\square(U, V) := \inf_{\text{mpb } \phi} \|U - V^\phi\|_\square. \tag{5}$$

If $\{\mathcal{G}_n\}$ converges in homomorphism densities to $\mathbf{W}$, then (up to relabeling)

$$\delta_\square(\overline{\mathbf{W}}_n, \mathbf{W}) \longrightarrow 0, \tag{6}$$

where the induced graphon $\overline{\mathbf{W}}_n$ is defined by $\overline{\mathbf{W}}_n(u,v) = \sum_{i,j=1}^n [\mathbf{W}_n]_{ij} \chi_{I_i}(u) \chi_{I_j}(v)$ with $I_i = [\frac{i-1}{n}, \frac{i}{n})$. Conversely, $\delta_\square(\overline{\mathbf{W}}_n, \mathbf{W}) \to 0$ implies $t(\mathcal{F}, \mathcal{G}_n) \to t(\mathcal{F}, \mathbf{W})$ for all $\mathcal{F}$.

Convergence of graph–feature pairs (with relabeling) Let $\mathbf{Z}_n \in \mathbb{R}^n$ be node features on $\mathcal{G}_n$ and define the induced feature function $\overline{\mathbf{Z}}_n(u) = \sum_{i=1}^n [\mathbf{Z}_n]_i \, \chi_{I_i}(u)$. We say $(\mathcal{G}_n, \mathbf{Z}_n) \to (\mathbf{W}, \mathbf{Z})$, with $\mathbf{Z} \in L^2(I)$, if there exists a sequence of mpb's $\{\phi_n\}$ such that

$$\delta_\square(\overline{\mathbf{W}}_n, \mathbf{W}) \to 0 \quad \text{and} \quad \|\overline{\mathbf{Z}}_n - \mathbf{Z}^{\phi_n}\|_{L^2(I)} \to 0. \tag{7}$$

Equivalently, for discrete permutations $\{\pi_n\}$ (with associated mpb's ϕ_{π_n} mapping $I_i \mapsto I_{\pi_n(i)}$), convergence holds when $\|\overline{\mathbf{Z}}_{\pi_n(\mathcal{G}_n)} - \mathbf{Z}\|_{L^2(I)} \to 0$ and $\delta_\square(\overline{\mathbf{W}}_{\pi_n(\mathcal{G}_n)}, \mathbf{W}) \to 0$. This establishes a permutation-invariant link between discrete graphs with features and their graphon limits [Medvedev, 2013, Ruiz et al., 2021].

A.2 Deterministic graphon sampling procedure

The reverse direction, constructing a sequence of discrete graphs that converges to a given graphon, plays an essential role in connecting theory with experiments. We adopt a standard *deterministic sampling procedure* that depends on whether the graphon is weighted or unweighted.

For the **unweighted** case, let $\mathbf{W} : I^2 \to \{0,1\}$ denote a binary graphon with support $\mathbf{W}^+ = \{(u,v) \in I^2 : \mathbf{W}(u,v) = 1\}$. A discrete unweighted graph $\mathcal{G}_n$ with n nodes is obtained by placing nodes at $u_i = i/n$ and defining the edge set

$$E_n = \{(i,j) \in [n] \times [n] : (I_i \times I_j) \cap \mathbf{W}^+ \neq \emptyset\}, \tag{8}$$

where $I_i = [\frac{i-1}{n}, \frac{i}{n})$. The adjacency matrix is then $[\mathbf{W}_n]_{ij} = 1$ if $(i,j) \in E_n$ and 0 otherwise.

For the **weighted** case, let $\mathbf{W} : I^2 \to I$ be a weighted graphon. The weighted adjacency matrix is obtained by direct evaluation,

$$[\mathbf{W}_n]_{ij} = \mathbf{W}\big(\tfrac{i-1}{n}, \tfrac{j-1}{n}\big), \qquad i,j \in [n]. \tag{9}$$

Let $\mathbf{Z} \in L^\infty(I)$ be a graphon feature function. The corresponding discrete node features are given by either averaging or direct evaluation on subintervals:

$$[\mathbf{Z}_n]_i = n \int_{I_i} \mathbf{Z}(u)\, du \quad \text{or} \quad [\mathbf{Z}_n]_i = \mathbf{Z}\big(\tfrac{i-1}{n}\big), \tag{10}$$

and we adopt the latter for simplicity.

Under this deterministic sampling, the sequence of graph–feature pairs $\{(\mathcal{G}_n, \mathbf{Z}_n)\}$ converges to the graphon–feature pair $(\mathbf{W}, \mathbf{Z})$ as $n \to \infty$, with the associated permutation equal to the identity.

A.3 Box counting dimension

The box counting dimension of a set S is defined in terms of $N(\epsilon)$, the number of boxes of common side length ϵ that covers the set:

$$\dim_B(S) = \lim_{\epsilon \to 0} -\frac{\log N(\epsilon)}{\log \epsilon} \tag{11}$$

For the Koch and hexaflake fractals considered in our work, the open set condition applies and box-counting dimension is equivalent to Hausdorff dimension. [Hutchinson, 1981]

The three binary graphons used in our experiments have box-counting dimension: HSBM ($b = 1$), Koch ($b = 1.26$), and Hexaflake ($b = 1.77$).

B Spectral Graph Neural Networks

A variety of successful GNN architectures have been proposed [Defferrard et al., 2016, Bruna et al., 2014, Niepert et al., 2016] but for our discussion we utilize spectral graph convolutional networks. [Kipf and Welling, 2016] This GNN architecture utilizes a graph shift operator $\mathbf{S}_n \in \mathbb{R}^{n \times n}$ to encode the graph topology, an approach common in graph signal processing. [Shuman et al., 2013, Sandryhaila and Moura, 2013, Ortega et al., 2018] We choose the node-normalized adjacency matrix as the graph shift operator, $\mathbf{S}_n = \frac{1}{n}\mathbf{W}_n$. For a graph feature $\mathbf{x} \in \mathbb{R}^{n \times F}$ and graph filter $\mathbf{h} = [h_0, h_1, \ldots, h_{K-1}]^\top \in \mathbb{R}^K$ acting on K-hop neighbors, the graph convolution operator is defined as:

$$\mathbf{h} *_{\mathbf{S}_n} \mathbf{x} := \sum_{k=0}^{K-1} h_k \mathbf{S}_n^k \mathbf{x} \tag{12}$$

We now develop an L-layer GNN. Denote the number of node features present in layer ℓ by F_ℓ. By $\mathbf{h}_\ell^{fg}$ we denote the graph filter transforming the g-th input feature into the f-th output feature in layer ℓ. We denote $\mathbf{\Theta} = \{\mathbf{h}_\ell^{fg} \in \mathbb{R}^K : f \in [F], g \in [F], \ell \in [L]\}$ as the set of all trainable filter parameters. The explicit GNN output for the f-th feature of the ℓ-th layer for $f \in [F]$ is:

$$\mathbf{X}_\ell^f = \rho\left(\sum_{g=1}^{F} \mathbf{h}_\ell^{fg} *_{\mathbf{S}_n} \mathbf{X}_{\ell-1}^g\right) \tag{13}$$

where ρ is a nonlinear activation function and $\mathbf{X}_0 = \mathbf{Z}_n$ are the input features. We compactly denote this GNN output as $\Phi_n(\mathbf{S}_n; \mathbf{Z}_n; \mathbf{\Theta}) = \mathbf{X}_L$. In our experiments, we choose $F = 1$.

C Convergence of discrete graph dynamics to graphon dynamics

We now formalize the connection between discrete graph dynamics and their continuous graphon limits. Let $\mathcal{G}_n = \langle V_n, E_n, \mathbf{W}_n \rangle$ denote a deterministic graph with adjacency weights $[\mathbf{W}_n]_{ij} = \mathbf{W}(\frac{i-1}{n}, \frac{j-1}{n})$, where $\mathbf{W} : I^2 \to I$ is a fixed graphon. We consider node features $[\mathbf{X}_n]_i(t)$ evolving according to discrete dynamics of the form

$$\frac{d}{dt}[\mathbf{X}_n]_i(t) = \frac{1}{n}\sum_{j=1}^{n}[\mathbf{W}_n]_{ij}\, F([\mathbf{X}_n]_j(t),\, [\mathbf{X}_n]_i(t)) + G([\mathbf{X}_n]_i(t))\,, \quad i = 1, \ldots, n, \tag{14}$$

where F encodes pairwise interactions and G represents nodewise reaction terms. The normalization factor $1/n$ ensures that (14) approximates an integral operator in the large-n limit.

Define the piecewise-constant interpolant $\mathbf{X}_n(t, u) = \sum_{i=1}^{n}[\mathbf{X}_n]_i(t)\, \chi_{I_i}(u)$, where $I_i = [\frac{i-1}{n}, \frac{i}{n})$. As $n \to \infty$, under suitable regularity conditions on F, G, and $\mathbf{W}$, the discrete system (14) converges to the graphon-limit dynamics:

$$\partial_t \mathbf{X}(u,t) = \int_0^1 \mathbf{W}(u,v)\, F(\mathbf{X}(v,t),\, \mathbf{X}(u,t))\; dv + G(\mathbf{X}(u,t))\,, \quad u \in I. \tag{15}$$

This formulation unifies a variety of graph-based dynamical systems within a single continuum framework. In particular, the discrete dynamics used in our experiments converge to their corresponding graphon-limit equations as summarized in Table 4.

Graphon Neural Differential Equations To study the infinite-node limit of Graph Neural Differential Equations (GNDEs) parametrizing by spectral GCN as described in section B, Yan et al. [2025] introduced the *Graphon Neural Differential Equation* (Graphon-NDE), formally defined as

$$\frac{\partial}{\partial t}\mathbf{X}(u,t) = \Phi(\mathbf{W}; \mathbf{X}(u,t); \mathbf{\Theta}(t)) \tag{16}$$

where Φ is a graphon neural operator acting through $\mathbf{W}$ with time-dependent parameters $\mathbf{\Theta}(t)$. This formulation can be interpreted as a neural integro-differential equation evolving on a continuum of nodes.

Solution convergence As the graph–feature pairs $(\overline{\mathbf{W}}_n, \overline{\mathbf{Z}}_n)$ converge to $(\mathbf{W}, \mathbf{Z})$ in the cut norm and $L^2(I)$, respectively (see Appendix A.1), the corresponding discrete ODE solutions $\mathbf{X}_n(\cdot, t)$ converge to the continuum graphon-limit solution $\mathbf{X}(\cdot, t)$ of (15). When the interaction and reaction functions F and G are globally Lipschitz and the trajectories remain bounded on $[0, T]$, the solution operator of (15) is Lipschitz-continuous with respect to perturbations in both the graphon and the initial feature function. Consequently, for Hölder-smooth graphons ($\mathbf{W} \in C^{0,\alpha}$) and feature functions ($\mathbf{Z} \in C^{0,\beta}$), the trajectories converge at rate $\mathcal{O}(n^{-\min\{\alpha,\beta\}})$. For binary graphons, the rate deteriorates to $\mathcal{O}(n^{-(1-\frac{b}{2})})$, where b denotes the box-counting dimension of the boundary of $\operatorname{supp}(\mathbf{W})$ Medvedev [2013]. A similar rate of convergence to (15) was recently established in [Yan et al., 2025] for continuous-depth spectral GCNs, demonstrating that GNDEs inherit the same asymptotic behavior in their trajectory-wise approximation to graphon-limit dynamics. We compare these theoretical rates to our observed empirical rates in Table 3.

Table 4: Discrete graph dynamics and their corresponding graphon limits. Normalization by $1/n$ ensures convergence as $n \to \infty$.

Dynamics	Discrete $\to$ Graphon Limit
Linear Heat	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_j w_{ij}\big([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\big) \;\Rightarrow\; \partial_t \mathbf{X}(u) = \int_0^1 W(u,v)\,\big(\mathbf{X}(v) - \mathbf{X}(u)\big)\,dv$
Nonlinear Heat	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_j w_{ij}\,\sin\big([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\big) \;\Rightarrow\; \partial_t \mathbf{X}(u) = \int_0^1 W(u,v)\,\sin\big(\mathbf{X}(v) - \mathbf{X}(u)\big)\,dv$
Allen–Cahn	$\frac{d}{dt}[\mathbf{X}_n]_i = -\epsilon^2 \frac{1}{n}\sum_j w_{ij}\big([\mathbf{X}_n]_j - [\mathbf{X}_n]_i\big) + [\mathbf{X}_n]_i - [\mathbf{X}_n]_i^3 \;\Rightarrow\; \partial_t \mathbf{X}(u) = -\epsilon^2 \int_0^1 W(u,v)\,\big(\mathbf{X}(v) - \mathbf{X}(u)\big)\,dv + \mathbf{X}(u) - \mathbf{X}(u)^3$
Nonlinear Consensus	$\frac{d}{dt}[\mathbf{X}_n]_i = \frac{1}{n}\sum_j w_{ij}\,\frac{[\mathbf{X}_n]_j - [\mathbf{X}_n]_i}{1 + ([\mathbf{X}_n]_j - [\mathbf{X}_n]_i)^2} \;\Rightarrow\; \partial_t \mathbf{X}(u) = \int_0^1 W(u,v)\,\frac{\mathbf{X}(v) - \mathbf{X}(u)}{1 + (\mathbf{X}(v) - \mathbf{X}(u))^2}\,dv$

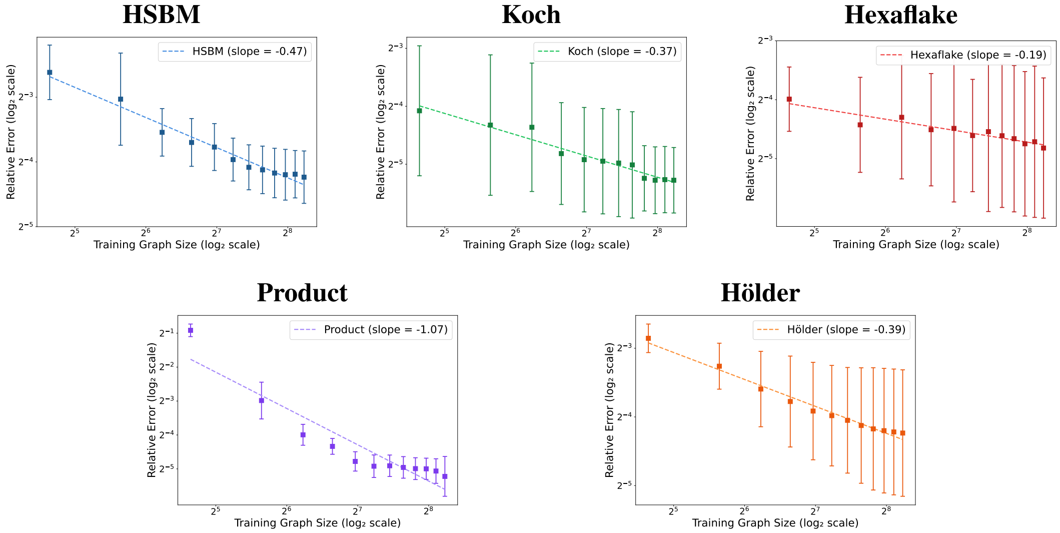

Figure 3: Log-log plots of mean relative L^2 error for the linear heat dynamics across ten trials per graphon, including line of best fit for convergence rate estimation. From top to bottom, left to right: HSBM, Koch, Hexaflake, Product, and Hölder results are reported in individual plots with standard deviation shown.

D Numerical experiment details

In Figure 3, we present the full relative L^2 error results for the data reported in Figure 2 with error bars shown for each graphon. We observe that graphons leading to slower rates of convergence also tend to exhibit higher variance, which may also be connected to similar structural considerations. Much of the variance in these figures is explained by the persistent presence of two or three outliers per ten runs, where no meaningful improvement in the models occurs as graph size increased. This phenomenon underscores the difficulty of studying transfer error in practical settings, as suboptimal training due to noise in model weight initialization and training set data can lead to vastly different behavior in the trendline. This decreasing error behavior was witnessed exclusively in the linear heat setting, while other dynamics quickly had errors decrease to a low but steady level. Models for the other dynamics were still able to effectively learn and transfer from graph sizes as low as 100, with an example shown in Figure 4.

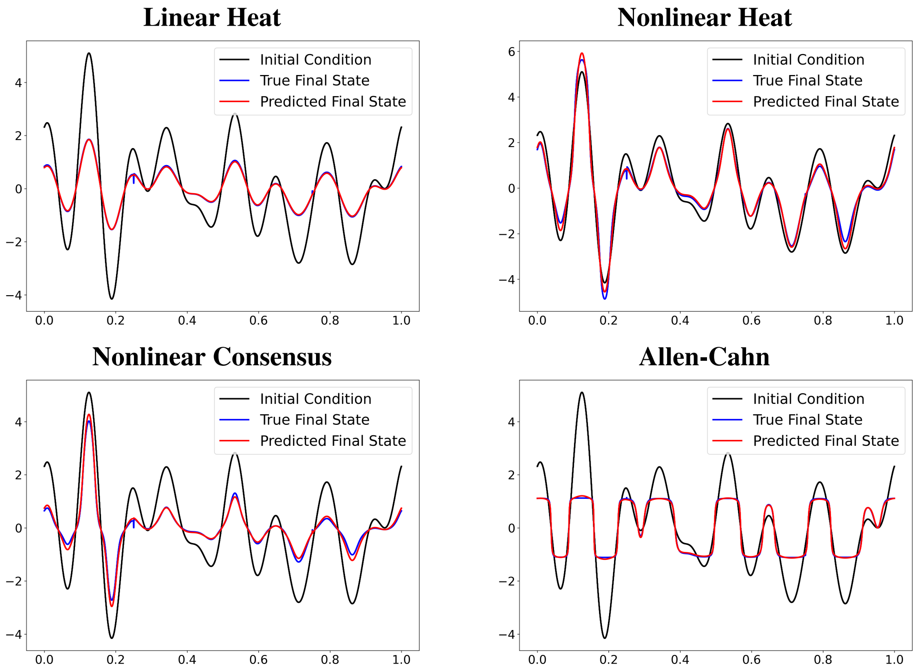

Figure 4: Initial condition (black) plotted with true final state of evolved dynamics (blue) and GNDE predicted final state (red). GNDE models shown here are trained on graphs of size 100 generated from the HSBM graphon and zero-shot transferred to a graph with size 1000. Top: Linear and nonlinear heat equations, $T = 2$ and $T = 4$ respectively. Bottom: Nonlinear consensus and Allen-Cahn with $\epsilon = \frac{1}{2}$, $T = 1$ and $T = 10$ respectively. All models show satisfactory performance with moderate errors occurring in the regions of rapid change, especially near the sharp boundary of the two graph regions at $x = \frac{1}{4}$ and $x = \frac{3}{4}$.

While the plots above show the ability of the model to learn the behavior of the dynamics satisfactorily, this does not fully answer the question of the plateau observed in the transfer rates. We show representative logarithmic loss curves in Figure 5 to allow qualitative comparison of linear and nonlinear loss behavior. The nonlinear loss tends to continue decreasing throughout the training interval, though the absolute magnitude of the decrease is small.

D.1 Implementation details

To choose GNDE model hyperparameters, we utilized a grid search across 10 different training seeds, training a model with the desired hyperparameters from scratch with the largest trained graph size of 300 for the HSBM graphon and linear heat dynamics. Average MSE on a newly generated test set was used as the metric for selection. Grid search choices and chosen values are reported in Table 5. Hyperparameter selections were confirmed to work satisfactorily across graphons and dynamics.

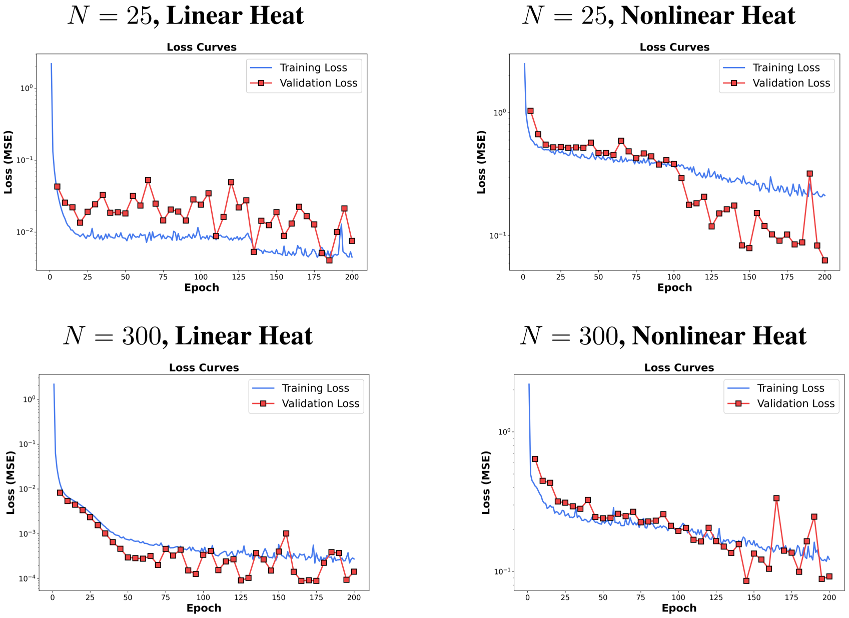

Figure 5: Representative logarithmic plots of training and validation loss for smallest and largest graph sizes of $N = 25$ and $N = 300$ for linear and nonlinear heat equations. Top: Loss curves for $N = 25$, with the linear heat equation left, and nonlinear right. Bottom: Loss curves for $N = 300$, with the linear heat equation left, and nonlinear right. Training loss is shown in blue and validation loss in red. Note that the y-axis is scaled logarithmically to show the fluctuations in loss clearly; the absolute scale of the loss change after the initial descent is very small.

Table 5: Hyperparameters used in grid search, both grid search choices and final selected values.

Hyperparameter	**Grid Search**	**Value**
Learning Rate	$\{10^{-2}, 10^{-3}, 10^{-4}\}$	10^{-3}
Epochs	{50, 100, 200}	200
K	{2, 3}	2
GNN Hidden Features	{1, 4, 8, 16, 32}	16

We further specify all training hyperparameter choices. We used the Adam optimizer Kingma and Ba [2014] with the standard hyperparameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for model training. We utilized the classical fourth-order Runge-Kutta solver [Runge, 1895, Butcher, 2008] for all GNDE evaluations, as it provides a favorable balance between computational efficiency and accuracy. The $\tanh$ activation function was used in all layers of the spectral GCN model (13). Dropout was considered, but significantly lowered test performance, and in final models no dropout was utilized. A very light early stopping criteria was implemented, using as a metric a plateau of performance in validation accuracy spanning more than 50 epochs. This rarely triggered in our experiments, as the observed convergence during the training of the models was fairly even and slow.

Computational complexity Let $\mathcal{G}_n$ be a discrete graph. The time complexity of forward computation for an L-layer GNDE with F node features and K-hop neighborhoods on $\mathcal{G}_n$ is $n_F \cdot \mathcal{O}(|E(\mathcal{G})|LKF + |V(\mathcal{G})|LKF^2)$, where n_F is the number of forward evaluations required by the ODE solver. For all graphons considered in our work and many real applications, the edge count term is dominant. The space complexity of GNDE evaluation is $\mathcal{O}(|E(\mathcal{G})|F + |V(\mathcal{G})|KF)$, with the first term due to propagation of information along edges typically being dominant, while node feature storage is often in a relatively low dimension.

All experiments were performed locally on 4 Nvidia A4000 GPUs, with all experimental results computed over two weeks of wall time. While no individual step of the training process is computationally prohibitive at these scales, training many models requires extensive feature generation and optimization, and as such remains a relatively slow task.