CASIN: CURVATURE-AWARE SIMPLICIAL NETWORKS FOR TIME-DEPENDENT PDES ON UNKNOWN RIEMANNIAN MANIFOLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Partial Differential Equations (PDEs) play a vital role in a wide range of scientific, engineering, and mathematical disciplines, as they describe complex phenomena involving functions of multiple variables and their rates of change. In many applications such as healthcare and Earth sciences, we need to develop a PDE solution, given only a finite number of irregularly located estimates of the system's states that live on an unknown manifold such as incipient caries on the teeth surface or irregularly spaced weather stations in areas with variable terrain. Despite that such scenarios are widely met in various domains, from wildfire analytics to biomedical imaging, numerical solutions of PDEs on non-Euclidean domains, including deep learning (DL)-assisted solvers, remain substantially under-investigated comparing to their Euclidean counterparts. We aim to address this gap and focus on time-dependent PDEs defined on unknown Riemannian manifolds, treating these manifolds as the underlying domain. To approximate and evaluate differential operators and the associated PDE solutions on the unknown manifold, we leverage the tools of simplicial convolution as the backbone for the DL-assisted solver. We develop a new flexible solver, Curvature-Aware SImplicial Networks for Time-Dependent PDEs on Unknown Manifolds (CASIN). We evaluate CASIN's utility for approximating the complex pattern-formation dynamics governed by nonlinear reaction–diffusion PDE systems on synthetic manifolds and tooth surfaces. Across 8 state-of-the-art neural PDE solvers and 9 settings, CASIN attains the lowest error in 7, reducing RMSE by upto 67% relative to the strongest curvature respecting method and by nearly an order of magnitude relative to solvers that are agnostic to geometry. On real tooth geometries and it is best on all three simulated reaction–diffusion systems. We further evaluate its performance on a nonlinear advection system modeling fluid dynamics on manifolds.

1 INTRODUCTION

Partial Differential Equations (PDEs) are fundamental to modeling a vast array of natural and engineered systems, from physics to economics to healthcare (Bertozzi et al., 2025), and have a long history in applied mathematics (Archibald et al., 2005). While extensive progress has been made in solving PDEs on Euclidean domains Adler et al. (2025), many real-world problems are inherently non-Euclidean, requiring the development of methods that can operate on complex geometries such as manifolds. For example, biologically active processes on a tooth occur on smooth, curved surfaces such as enamel and gingiva, that are locally Euclidean but globally non-flat and embedded in three dimensions (Fabregas & Rubinstein, 2014). PDEs on such tooth manifolds allow us to model mineral loss during dental caries, while better accounting for essential individual anatomical constraints. Another immediate application would be wildfire spread models. Wildfire spread propagates along the Earth's curved surface rather than in flat space, and PDEs on a terrain manifold deliver more realistic characterization of uphill acceleration and essential barrier effects that PDEs on Euclidean domains are likely to miss (Nieding et al., 2025). In turn, solving PDEs on unknown manifolds remains a challenging and still substantially less investigated (comparing to the Euclidean domain) problem at the interface of applied mathematics, differential geometry and a large corpus of applications of sciences and engineering (Liang et al., 2024). The main issue in this fundamental computational

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

problem is in the approximation and evaluation of differential operators and the PDE solution on the unknown manifold, given typically only a finite number of randomly distributed observed points.

Recent advances in machine learning (ML), particularly, geometric and topological deep learning (Papamarkou et al., 2024) have opened up new possibilities for solving PDEs on complex, non-flat domains. However, in contrast to a wide range of ML-assisted PDE solvers on Euclidean domains, including both families of Physics Informed Neural Networks (PINNs) and Neural Operators (NOs), integration of ML for PDE solutions on unknown Riemannian manifolds still remains a largely uncharted territory (Ryu et al., 2025; Liao et al., 2025).

Here we invoke the emerging machinery of simplicial convolution, rooted in discrete exterior calculus, as the backbone for DL-assisted solvers for time-dependent PDEs on an unknown Riemannian manifold, without known parameterizations or mesh and given only a set of irregularly located observations of the system's state at multiple time steps. Our key novelty extends previous approaches Iakovlev et al. (2021) for PDEs on Euclidean domains. CASIN is based on curvature-aware architecture that utilizes information from the local geometric structure, allowing the model to dynamically adapt its computations based on curvature and other geometric features. We demonstrate the utility of the proposed curvature-aware simplicial neural network (CASIN) in application to modeling the complex dynamics described by highly nonlinear reaction–diffusion systems of PDEs such as the Fisher–Kolmogorov–Petrovsky–Piskunov model, the Gray-Scott model, and the Turing Pattern formation on various synthetic manifolds. We further implement CASIN to nonlinear advection systems, specifically shallow water systems, to illustrate its robustness in capturing spatial dynamics of flow velocity fields. Based on our experiments, CASIN is able to more effectively capture differential operators with additional spatial dynamics stemming from inhomogeneous systems with advection and reaction terms, demonstrating CASIN can handle complicated spatial dynamics. Furthermore, motivated by the problem of developing a patient-specific digital twin for dental diseases, we demonstrate CASIN on simulated reaction–diffusion dynamics over real molar geometries, a first step toward patient-specific dental models.

The significance of our contributions can be summarized as follows:

- We propose CASIN, the first neural PDE solver that integrates curvature-aware features with simplicial neural networks for unknown Riemannian manifolds represented solely by point clouds.
- We evaluate CASIN performance on a wide range of biologically-motivated and fluid dynamics systems of PDEs across synthetic manifolds, showing its competitive capabilities with respect to the state-of-the-art neural PDE solvers.
- We show that CASIN provides a flexible framework for modeling dental decay dynamics on patient-specific tooth surfaces by solving PDEs directly on highly curved dental manifolds, addressing a largely unexplored application domain of physics based machine learning models for dental applications.

2 Related work on Neural PDE Solvers on Manifolds

In the last couple of years *Physics-Informed Neural Networks (PINNs)* on manifolds have emerged as a powerful alternative to more conventional numerical methods for PDEs on non-Euclidean domains, but still remain substantially under-studied with respect to their Euclidean counterparts. (For an overview non-ML based numerical methods for PDEs on manifolds see Appx. A.) Most such PINNs are limited to the scenarios where either the manifold is known or well parametrized or where the PDE is defined on a structured grid.

In turn, Δ-PINN of Costabal et al. (2024) considers an encoding mechanism for PINNs based on the eigenfunctions of the Laplace-Beltrami operator on meshed manifolds, and a related spectral approach is undertaken by Zelig & Dekel (2023) under the grid free settings but restrictive conditions on local coordinates. The approach of Lei et al. (2025) focuses on theoretical guarantees for a mesh-free CNN-based PINN but assumes that the manifold is known (i.e, sphere). Similarly, Shallow PINN of Hu et al. (2024) requires knowledge of explicit surface information. $\mathcal{M}$-PINN of Valperga et al. (2025) also discuss a mesh-free PINN, where the manifold is first partitioned into local overlapping patches (charts) and then separate PINN is trained for each individual chart, which implies higher

computational costs and potential loss of information as only dependence across overlapping patches is considered. Finally, the most recent approach of Ryu et al. (2025) proposes PINN for solving PDEs on point clouds via surface reconstruction. While arguably the method of Ryu et al. (2025) is the most flexible in terms of its assumptions on PDE and its domain among all existing PINNs on manifolds.

As such, most existing PINNs either do not explicitly account for the geometry of the non-Euclidean domain where the problem is being solved or impose various restrictive assumptions on the domain structure. This in turn limits the utility of these approaches under practical scenarios where the unknown non-Euclidean domain is represented only by a raw point cloud and there are no geometric priors as, for example, in the cases of dental crown designs (Hosseinimanesh et al., 2025), computer-assisted surgery (Cotin et al., 2024), patient-specific brain modeling (Zapf et al., 2022) and other biomedical imaging applications (Banerjee et al., 2024). To the best of our knowledge, until now, only Ryu et al. (2025) have considered PINN for solving PDEs on point clouds without requiring any geometric priors.

In turn, the alternative paradigm of *Neural Operators (NO)* focuses on developing DL models that learn the solution operator of a PDE via designing a map between function spaces instead of functions Karniadakis et al. (2021); Kovachki et al. (2023); Jha (2025); Wang & Townsend (2025). Canonical NO examples include DeepONets (Lu et al., 2021) and Fourier neural operators (Li et al., 2021). More recently, topological methods have been successfully applied to NO (Zheng et al., 2026), however, they assume mesh structure to be given which is often absent in real world cases making it a key limitation of existing work. While being a very powerful machinery, NOs typically require a substantial amount of training data on a grid. To the best of our knowledge, CurvGT (Liao et al., 2025) is the only NO for PDEs on a manifold on point-cloud samples.

CASIN specifically differs from two approaches highlighted above. SNN-PDE learns PDE dynamics with simplicial neural networks but was designed for and evaluated on Euclidean manifolds. CurvGT builds curvature into graph attention for PDEs on manifolds but works only with pairwise graph connections, without accounting for higher-order structural characteristics. To the best of our knowledge, CASIN is the first to bring simplicial learning to PDEs on unknown manifolds. It reconstructs a simplicial complex directly from the point cloud and combines it with curvature and edge and triangle geometry. This makes our approach (which is formally neither PINN nor NO) the first to advance PIML methodology by harnessing graph and simplicial learning for PDE solvers on unknown Riemanninan manifolds represented by raw point clouds.

3 Problem Statement: Time-Dependent PDEs on Unknown Riemannian Manifolds

The solution of PDEs on unknown Riemannian manifolds requires advancing analytical and numerical techniques to curved and non-Euclidean geometries. Unlike flat Euclidean spaces, where global Cartesian coordinates and linear structures substantially simplify analysis, Riemannian manifolds demand the use of local coordinate charts, intrinsic differential operators, and geometric tools that respect the underlying curvature. Numerical methods must also be carefully generalized, since discretization on curved domains involves constructing meshes or mesh-free approximations that conform to the manifold's topology and geometry.

Problem Statement: Let $\mathcal{M} \subset \mathbb{R}^d$ be a compact Riemannian manifold, where the underlying geometry is characterized by a smooth metric tensor g. Under this setting, classical PDEs are generalized to PDEs on manifolds by replacing standard Euclidean differential operators with their intrinsic counterparts that respect the geometry of $\mathcal{M}$. In particular, let n be a normal vector to $\mathcal{M}$, then the surface gradient $\nabla_{\mathcal{M}} u = \nabla u - (\nabla u \cdot n)n$ and the Laplace–Beltrami operator $\Delta_{\mathcal{M}} = \nabla_{\mathcal{M}} \cdot \nabla_{\mathcal{M}}$ are the natural extensions of the Euclidean gradient and Laplacian, respectively, incorporating the geometry of $\mathcal{M}$. Armed with this geometric framework, we can now analyze PDEs directly on curved domains, allowing us to systematically address various applications involving data living on non-Euclidean spaces.

Motivated by the questions arising in biomedical digital twins that need to account for changing and irregular geometry, we present an analysis using time-dependent PDEs on unknown curved domains.

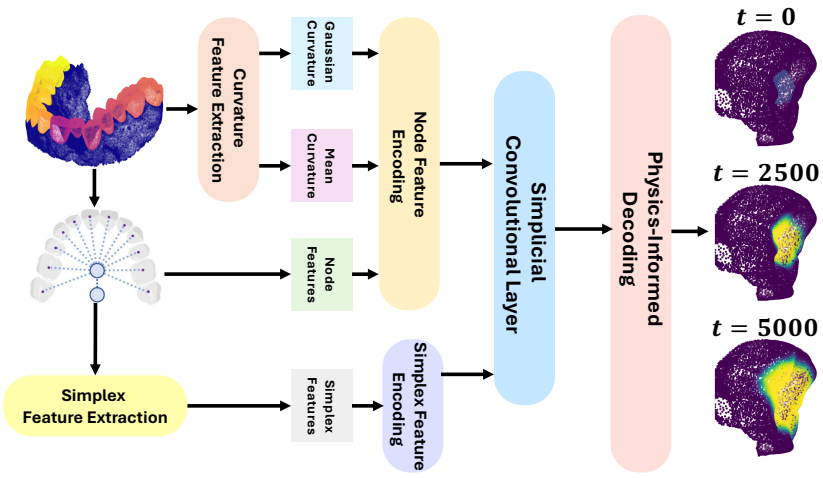


Figure 1: Overall framework of CASIN.

Let $\Omega \subset \mathcal{M}$ be a non-empty, compact, relatively open subset, $u : \mathcal{M} \to \mathbb{R}$ be a twice continuously differentiable function on $\mathcal{M}$, with u_t being the time derivative of u. Furthermore, in light of the nature of the motivating applications in the healthcare domain, we assume that the manifold $\mathcal{M}$ is unknown and, moreover, we are given only a set observations $\{\mathbf{y}(t)\}_{t=0}^{M} \in \mathbb{R}^{N\times(M+1)}$ of the system's state $\mathbf{u}(t) = \big(u(\mathbf{x}_1, t), \ldots, u(\mathbf{x}_N, t)\big)'$ at N arbitrarily located points $\{\mathbf{x}_j\}_1^N$ over manifold $\mathcal{M}$ and $(M+1)$ time steps $(t_0, \ldots, t_M)$. Such a framework may correspond to, for example, $\mathcal{M}$ being an organ characterized by intricate anatomical structure and $\{\mathbf{x}_j\}_1^N$ being locations of biopsies.

Our goal is then to find an approximation $\tilde{u}(F, x, t)$ to the solution of the following time-dependent PDE on $\mathcal{M}$, using observations $\{\mathbf{y}(t)\}_{t=0}^{M}$ recorded over arbitrary locations $\{\mathbf{x}_j\}_1^N$ over $\mathcal{M}$ and time steps $\{t_k\}_{k=0}^{M}$:

$$\begin{cases} u_t(t,x) = F(x, u(t,x), \nabla_{\mathcal{M}} u(t,x), \Delta_{\mathcal{M}} u(t,x)), \\ \forall x \in \mathcal{M},\ t \in [0,T], \\ G(x, u(t,x), \nabla_{\mathcal{M}} u(t,x), \Delta_{\mathcal{M}} u(t,x)) = 0, \\ x \in \Omega,\ t \in [0,T]. \end{cases} \tag{1}$$

with initial conditions $u(0, x) = H(x)$.

To solve PDE on an unknown $\mathcal{M}$, we need to start from reconstructing the intrinsic properties of $\mathcal{M}$ from a sample of points $\mathcal{M}$ $\{\mathbf{x}_j\}_1^N$. Suppose we observe a set of irregularly spaced locations $\mathcal{P} = \{x_j\}_{j=1}^N \subset \mathbb{R}^3$ on a smooth unknown Riemannian manifold $\mathcal{M}$ which is endowed with the Riemannian metric g, induced by its embedding into the ambient space $\mathbb{R}^3$. To recover the manifold structure from this discrete representation, we first estimate the tangent space $T_p\mathcal{M}$ at each point p. Once the tangent plane is determined, we project the neighboring points onto this subspace, thereby reducing the local geometry to a lower-dimensional Euclidean setting. Within this projected neighborhood, we construct a triangulation, for example via Delaunay triangulation, to encode the local connectivity of the manifold. By applying this procedure consistently across the point cloud $\mathcal{P}$ and reconciling overlaps between neighborhoods, we obtain a global simplicial complex $\mathcal{K}$ that provides a combinatorial approximation of $\mathcal{M}$.

4 CASIN Architecture

The Curvature-Aware Simplicial Network (CASIN) integrates geometric information from Riemannian manifolds with higher-order topological features. The architecture consists of three main components: (i) curvature-aware feature encoding, (ii) simplicial convolution layers with boundary-aware message passing, and (iii) flow-map decoding for PDE and its solution approximation. We also provide detailed analysis of computational complexity and wall-clock times for CASIN in Appendix D along with reproducibility details.

4.1 CURVATURE-AWARE FEATURE ENCODING

Building upon the geometric quantities estimated in previous section, we encode both the local curvature information and the PDE state variables into neural network features.

Curvature Features: For each point $p \in P$, we estimate the metric tensor g via some suitable curve fitting, e.g., local polynomial or radial basis function interpolation, and subsequently compute the Riemann curvature tensor $R^{\rho}_{\sigma\mu\nu}$. For computational efficiency in the neural network, we extract scalar curvature invariants from R: the Gaussian curvature K and H from 2. These quantities capture the intrinsic and extrinsic geometry of the manifold, respectively, and are computed as:

$$K = \frac{R_{1212}}{\det(g)}, \quad H = \frac{1}{2} g^{ij} h_{ij} \tag{2}$$

where h_{ij} denotes the second fundamental form components derived from the local parameterization $z(u, v) = (u, v, h(u, v))^T$.

Node Feature Encoding: The initial node features combine the PDE state variables with the estimated curvature information:

$$\mathbf{h}_i^{(0)} = \text{MLP}_{\text{enc}}([U_i, K_i, H_i]) \in \mathbb{R}^d \tag{3}$$

where $U_i = (u^q(x_i))_{q=1}^{N_{eq}}$ with U being the solutions to the given PDE system with N-equations evaluated at point x_i, and d is the hidden dimension.

Simplex Feature Encoding: Following procedure in Appendix B.1, we extract geometric features for higher-order simplices. For edges (1-simplices), the feature vector x_e encodes:

$$x_e = [\ell_e, \hat{\mathbf{d}}_e] \in \mathbb{R}^4 \tag{4}$$

where $\ell_e = \|x_j - x_i\|$ is the edge length and $\hat{\mathbf{d}}_e = (x_j - x_i)/\ell_e$ is the unit direction vector. For triangles (2-simplices), the feature vector x_{tri} encodes:

$$x_{tri} = [A_{tri}, \hat{\mathbf{n}}_{tri}] \in \mathbb{R}^4 \tag{5}$$

where A_{tri} is the triangle area computed via $A_{tri} = \frac{1}{2}\|(x_j - x_i) \times (x_k - x_i)\|$ and $\hat{\mathbf{n}}_{tri}$ is the unit normal vector.

4.2 SIMPLICIAL CONVOLUTION LAYER

Our simplicial convolution layer performs message passing across simplex dimensions while respecting the boundary structure of the simplicial complex $\mathcal{K}$. We employ Hodge diffusion mechanism via the discrete Hodge Laplacians

$$L_0 = B_1 B_1^{\mathsf{T}}, \qquad L_1 = B_1^{\mathsf{T}} B_1 + B_2 B_2^{\mathsf{T}}, \tag{6}$$

where B_1 (vertex–edge) and B_2 (edge–triangle) are the boundary operators. By design, these are unweighted Laplacians that encode the higher order connectivity. Note that L_1 retains both the lower ($B_1^{\mathsf{T}} B_1$) and upper ($B_2 B_2^{\mathsf{T}}$) terms, so that although message passing is carried on vertices and edges (where metric information enters through curvature features), edge features remain coupled to the triangles through the upper term. Each layer applies L_0 and L_1 for intra-dimension diffusion and uses B_1 for boundary-aware message passing between vertices and edges, with residual connections and learnable MLPs. For more detailed formulation of simplicial convolution we refer readers to Ebli et al. (2020), while the background on Hodge-Laplacians is discussed in Appendix B.

4.3 FLOW-MAP DECODING

CASIN learns a discrete flow-map directly from observed trajectories such that PDEs remain unknown. After L simplicial convolution layers are applied, the node representations $\mathbf{h}_L$ are decoded into a residual update that advances the state by one step:

$$\hat{U}_{t+\Delta t} \;=\; U_t \;+\; \text{MLP}_{\text{dec}}\big(\mathbf{h}^{(L)}\big), \tag{7}$$

where $U_t = (u^q(\cdot, t))_{q=1}^{N_{\text{eq}}}$ is the observed state at the current step. Multi-step forecasts are produced by iteratively applying equation 7, $\hat{U}_{t+k\Delta t} = \Phi_\theta^k(U_t)$. The network receives only the state U_t and the estimated geometric features (K, H), i.e., the network is given neither the operator F nor its reaction terms, and it does not decode a Laplace–Beltrami term or symbolically separate diffusion from reaction. The geometry of $\mathcal{M}$ enters **solely** through the curvature features and the Hodge Laplacians built on the reconstructed complex.

The parameters θ are trained end-to-end to minimize the trajectory prediction error over the observed points $\{x_j\}_{j=1}^N$:

$$\mathcal{L}(\theta) \;=\; \sum_{t=1}^{T} \sum_{j=1}^{N} \left\| \tilde{u}_\theta(x_j, t) - y_j(t) \right\|_2^2, \tag{8}$$

where $\tilde{u}_\theta$ is the CASIN forecast obtained by iterating equation 7 from the initial state and $\{y(t)\}_{t=1}^M$ are the observations.

5 Experiments

Given the formulation above, CASIN provides a unified framework that both resolves the PDE dynamics and accounts for the geometric structure of the unknown manifold domain. To demonstrate this capability, we select three representative reaction-diffusion systems of PDEs, known for capturing rich and intricate pattern formations driven by nonlinear reaction dynamics. In addition, we also select a shallow water system of PDEs from fluid dynamics modeling, i.e., a nonlinear system of hyperbolic conservation laws, in order to test CASIN's capacity to capture velocity fields induced by nonlinear advection terms. These systems are intended to evaluate CASIN on its decomposition framework to recover the essential structure of the underlying dynamical behavior on the unknown manifolds. In particular, we employ three reaction-diffusion systems arising from modeling of biological pattern formations Murray (2002): the **Fisher-KPP** equation (9), the **Gray-Scott** system (10) and **Turing pattern** (11). These PDEs have been extensively investigated in biological modeling, for example, Fisher-KPP for invasive cell populations and tumor growth, Gray-Scott for morphogenesis and cellular differentiation, and Turing patterns for skin pigmentation and developmental patterning (Mimura et al., 2000; Ali & Saleem, 2023).

Fisher-KPP Equation:

$$u_t = D\Delta_{\mathcal{M}} u + ru(1-u) \tag{9}$$

where D is the diffusion coefficient and r is the growth rate.

Gray-Scott System:

$$\begin{aligned} u_t &= D_u \Delta_{\mathcal{M}} u - uv^2 + F(1-u) \\ v_t &= D_v \Delta_{\mathcal{M}} v + uv^2 - (F+k)v \end{aligned} \tag{10}$$

where F and k control the feed and kill rates.

Turing Pattern Formation:

$$\begin{aligned} u_t &= D_u \Delta_{\mathcal{M}} u + f(u, v) \\ v_t &= D_v \Delta_{\mathcal{M}} v + g(u, v) \end{aligned} \tag{11}$$

with reaction kinetics f and g satisfying the Turing instability conditions. For example, for our experiments we have used $f(u, v) = a - bu + \frac{u^2}{v}, \quad g(u, v) = u^2 - cv, v \neq 0$, which are the Gierer-Meinhardt equations, a well-known Turing system for biological processes (Gierer & Meinhardt, 1972).

We note that training for CASIN is performed under the assumption that PDE is unknown and we only have observation trajectory available.

On fluid dynamics, we consider a **Shallow Water System** Carlino & Gaburro (2023), i.e. a nonlinear system of time-dependent hyperbolic PDEs, describing the conservation of mass and momentum in a thin fluid layer on a manifold (see Appendix C for the detailed results).

We evaluate CASIN on both synthetic and real-world manifold datasets: torus, distorted torus, Stanford bunny (Turk & Levoy, 1994), and dental surfaces from the Teeth3DS dataset (Ben-Hamadou et al., 2022) (see more details in Appendix C.3).

Baselines We compare against eight state of the art neural PDE solvers. PINNsFormer (Zhao et al., 2024), PINN-Mamba (Xu et al., 2025), P^2INN (Cho et al., 2024), bts-PINN (Saleh et al., 2024), and SWIM-PDE (Datar et al., 2024) differ in how they parameterize the solution and optimise the residual, but all treat the domain as flat and hence are curvature agnostic. The remaining methods encode domain structure explicitly and isolate the two ingredients CASIN combines: SNN-PDE (Choi et al., 2024) shares our simplicial machinery but acts on Euclidean manifolds and does not share our curvature features, GLNO (Tang et al., 2025) is geometry-aware through the Laplace–Beltrami spectrum, and CurvGT (Liao et al., 2025) builds curvature into graph attention and previously reported the strongest published results for PDEs on manifolds. All learned baselines are trained under the identical observations-only state trajectories, no governing equations. Residual-based PINN baselines, which require the equation, are given the known PDE.

	Fisher KPP			Gray Scott			Turing Pattern		
Model	**Torus**	**Dist Torus**	**Bunny**	**Torus**	**Dist Torus**	**Bunny**	**Torus**	**Dist Torus**	**Bunny**
SNN-PDE(2024)	0.54±3e-2	0.58±3e-2	0.47±5e-2	0.52±5e-2	0.48±3e-2	0.41±3e-2	0.45±5e-2	0.51±8e-2	0.53±4e-2
bts-PINN(2024)	0.59±8e-2	0.67±1e-1	0.56±9e-2	0.66±6e-2	0.54±5e-2	0.53±8e-2	0.49±5e-2	0.55±1e-1	0.54±1e-1
PINNsFormer(2024)	0.45±5e-2	0.49±5e-2	0.51±4e-2	0.51±4e-2	0.41±5e-2	0.46±4e-2	0.41±4e-2	0.49±5e-2	0.43±1e-1
SWIM-PDE(2024)	0.22±2e-2	0.31±4e-2	0.23±2e-2	0.28±2e-2	0.31±3e-2	0.19±3e-2	0.25±3e-2	0.21±5e-2	0.20±3e-2
P^2 INN(2024)	0.33±7e-2	0.41±7e-2	0.38±5e-2	0.29±4e-2	0.34±5e-2	0.33±5e-2	0.25±6e-2	0.27±3e-2	0.33±8e-2
PINN-Mamba(2025)	0.42±9e-2	0.49±9e-2	0.37±7e-2	0.33±7e-2	0.45±9e-2	0.32±4e-2	0.36±7e-2	0.40±9e-2	0.44±1e-1
CurvGT(2025)	**0.02±4e-4**	0.04±7e-3	0.03±5e-3	**0.02±1e-4**	0.05±1e-2	0.03±9e-3	0.06±8e-3	0.04±8e-3	0.05±7e-4
GLNO(2025)	0.07±1e-2	0.12±3e-2	0.15±2e-2	0.06±9e-2	0.08±1e-2	0.10±2e-2	0.14±3e-2	0.11±4e-2	0.21±4e-2
CASIN (Ours)	0.03±2e-4	**0.03±6e-3***	**0.02±3e-3***	0.03±7e-4	**0.03±8e-4****	**0.01±1e-3****	**0.04±9e-3***	**0.02±5e-3****	**0.03±8e-3***

Table 1: RMSE for Time Dependent PDEs on Synthetic Manifolds; best results are in **bold**, and the 2nd best are underlined; * and ** denote significance (p-val. < 0.1) and stat. sign. (p-val. < 0.05), respectively.

	Components			Distorted Torus			Dental Molar		
Model	L_0	L_1	Curv.	Fisher-KPP	Gray-Scott	Turing	Fisher-KPP	Gray-Scott	Turing
CASIN with L_0	✓			0.09± 5e-2	0.05 ±2e-2	0.06 ±3e-2	0.033 ±1e-3	0.038 ± 2e-3	0.023 ± 3e-2
+ Edges (L_1)	✓	✓		0.07 ± 4e-2	0.05 ± 7e-3	0.04 ± 1e-2	0.026 ± 2e-3	0.029 ± 3e-3	0.018 ± 2e-3
CASIN (full)	✓	✓	✓	**0.03**±6e-3	**0.03**±8e-4	**0.02**±5e-3	**0.017**±1e-3	**0.021**±2e-3	**0.011**±2e-3

Table 2: Ablation over CASIN's components. RMSE (lower is better) as operators are added cumulatively: the vertex Laplacian L_0 alone, plus the edge Laplacian L_1, plus curvature features (K and H).

6 FINDINGS

To provide a comprehensive empirical analysis of CASIN, we compare it against 8 state-of-the-art baselines using 4 time-dependent PDE's across 3 synthetic manifolds, and 1 real-world dental manifold, where CASIN is tested in a real world scenario of irregularly curved surfaces through a case study on dental scans. We then ablate the model to isolate the contribution of its topological and geometric components and assess manifold recovery capabilities, along with the sensitivity analysis. *Across 8 baselines and 9 settings, CASIN attains the lowest Root Mean Square Error (RMSE) in 7 settings and is runner-up in the remaining two, with relative error reductions of up to* 67%. All experimental details, ensuring reproducibility are presented in Appendix C.

6.1 RESULTS ON SYNTHETIC MANIFOLDS

Table 1 reports RMSE for three reaction-diffusion systems on three manifolds of increasing geometric irregularity. CASIN is the best in 7 of the 9 settings and 2nd best in the other two, lowering mean RMSE from 0.038 to 0.027 relative to CurvGT, the strongest prior method. The margin over solvers that are agnostic to geometry improves greatly for example, against, SWIM-PDE, which has an RMSE of 0.244. From our experiments, we observe that geometry alone does not explain all of our gains. GLNO, which is also manifold-aware, still averages 0.116. The improvements are especially

noticeable on irregular geometries, the distorted torus and the Stanford bunny, where CASIN is best in all 6 settings, with relative reductions of $25\% - -67\%$ with three being statistically significant. Both of CASIN's losses occur on the standard torus, whose curvature is smooth and uniformaly varying, where CurvGT leads by 0.01 on Fisher–KPP and Gray–Scott. The exception is the Turing system, where CASIN leads on the torus as well (0.04 vs. 0.06). We attribute this to the higher order simplex features that CASIN accounts for. Our ablation study 7 isolates the effect directly.

Model	Fisher-KPP	Gray-Scott	Turing Patterns
SNN-PDE (2024)	0.276±4e-2	0.372±3e-2	0.199±1e-2
bts-PINN (2024)	0.341±2e-2	0.398±3e-2	0.153±8e-2
PINNsFormer (2024)	0.317±5e-2	0.381±7e-2	0.222±7e-2
SWIM-PDE (2024)	0.253±2e-3	0.194±2e-2	0.257±3e-2
P^2INN (2025)	0.178±5e-2	0.201±3e-2	0.125±9e-2
PINN-Mamba (2025)	0.231±1e-2	0.239±1e-2	0.109±3e-3
CurvGT (2025)	0.018±2e-3	0.047±1e-3	0.013±2e-3
GLNO (2025)	0.038±4e-3	0.132±2e-3	0.048±4e-3
CASIN (Ours)	**0.017**±1e-3	**0.021**±2e-3	**0.011**±2e-3

Table 3: RMSE for Time Dependent PDEs on Dental Manifolds for Fisher-KPP, Gray-Scott and Turing Patterns; best results are in bold

6.2 Tracking Tooth Decay with PDEs on Dental Surface

We now turn to dental manifolds, i.e., the area where the utility of neural solvers for PDEs tools remains largely unexplored. Modeling biological processes on dental surfaces necessitates the formulation of PDEs directly on curved manifolds, as the tooth enamel, where bacterial biofilm formation and carious demineralization occur, is inherently a two-dimensional surface embedded in three-dimensional space (Zheng et al., 2022; Yang et al., 2020). In dental context, the variables admit natural interpretations where, for example, u in Fisher-KPP may represent bacterial density or the extent of demineralization, with D capturing surface motility. We use the Teeth3DS dataset (Ben-Hamadou et al., 2022) which comprises 1,800 intraoral 3D scans containing 23,999 annotated teeth collected from 900 patients and validated by orthodontists. The data provides per-vertex annotations encoding tooth identity. This vertex-level labeling enables us to estimate higher order features for different dental regions and types, preserving intricate geometric features, fissures, cusps, and interproximal contact areas, that are critical for modeling bacterial colonization and caries progression. For our experiments, we sample points from individual tooth surfaces for molars from each scan by masking vertices according to their annotation labels and filtering faces to maintain manifold consistency. The initial bacterial colonies are seeded in well-connected vertices (degree > 3) within the 5th percentile of the distance from the centroid of the tooth, to ensure bacterial colonization in susceptible regions such as fissures and contact points. The Teeth3DS dataset also provides an official train/test split used in the 3DTeethSeg Challenge: 1,200 scans (600 patients) for training and 600 scans (300 patients) for testing. Table 3 shows the results on real-world dental manifolds from the Teeth3DS dataset. CASIN achieves the lowest RMSE in all cases, suggesting CASIN potential to effectively handle the complex, highly curved geometry of dental surfaces.

7 Ablation Study

We ablate CASIN by adding operators cumulatively starting from the vertex Laplacian L_0 to edges (L_1), to the full model with curvature features (CASIN). The L_0-only variant serves as a graph-only baseline, similar to a k-NN graph, it passes messages only between pairs of neighboring points and uses neither triangles nor curvature. All variants share the same backbone and training budget. We evaluate on two manifolds, a Distorted Torus and on Dental molar dataset as they have complex curvatures, each with three reaction–diffusion systems (Fisher–KPP, Gray–Scott, Turing); Table 2 reports prediction error (lower is better). Both components help; adding L_1 lowers the error by $\sim 20\%$ on average by supplying edge-based flux information that a node-only model cannot represent. Curvature gives the largest gain, a further $\sim 41\%$, improving every setting, which is consistent with reaction-diffusion equations being governed by local geometry (Murray, 2002). Overall, CASIN approximately halves the baseline error and is best in all the settings, showing that the topological

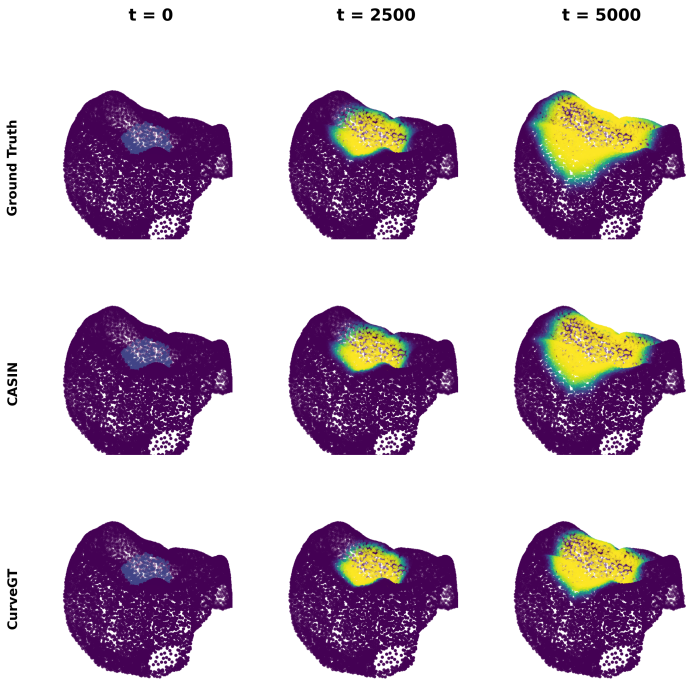


Figure 2: Comparison of CASIN and CurvGT with ground truth on a Molar sampled from Teeth3DS dataset.

(L_1) and geometric (curvature) components are complementary. In Appendix C, we analyze manifold recovery quality and provide an analysis of CASIN's performance and sensitivity analysis with respect to various curvature estimation approaches. The results suggest that in many cases the simple, yet computationally efficient polynomial curve learning may be the preferred choice to balance performance vs. computational costs.

8 Discussion, Limitations and Future Directions

We have presented CASIN, which to the best of our knowledge is the first curvature aware neural PDE-solver for systems of time dependent PDEs on unknown Riemanian manifolds which uses SNN to incorporate metric information. By leveraging topological deep learning tools, CASIN has demonstrated that combining curvature information with topological structure (simplicial complexes) yields robust and accurate forecasts on irregularly sampled surfaces. A current limitation of CASIN is the treatment of boundary conditions on manifolds with nontrivial or unknown boundaries. The simplicial representation naturally captures intrinsic geometric interactions in the interior of the manifold, but enforcing prescribed boundary constraints needs explicit identification of boundary simplices and modification of message-passing operations near the boundary, which we leave for future research. Another important, and potentially fruitful research directions relate to investigating the utility of CASIN to pre-condition NO approaches and serve as a backbone for PDE transfer learning.

AI USE STATEMENT

The authors used LLMs only for language editing and coding/debugging assistance, not for experimental design, result generation, or scientific ideas and analyses.

ETHICS STATEMENT

This works supports scientific modeling by solving time-dependent PDEs on complex geometries, with potential applications in biomedical and environmental research. We do not anticipate direct negative societal impacts from this work.

REPRODUCIBILITY STATEMENT

We provide all our code at `https://anonymous.4open.science/r/CASIN-DB06/README.md`. Further more, appendices C and D cover more details on reproducibility.

REFERENCES

James H Adler, Hans De Sterck, Scott MacLachlan, and Luke Olson. *Numerical partial differential equations*. SIAM, 2025.

Ishtiaq Ali and Maliha Tehseen Saleem. Spatiotemporal dynamics of reaction–diffusion system and its application to turing pattern formation in a gray–scott model. *Mathematics*, 11(6):1459, 2023.

Tom Archibald, Craig Fraser, and Ivor Grattan-Guinness. The history of differential equations, 1670–1950. *Oberwolfach reports*, 1(4):2729–2794, 2005.

Mukund Balasubramanian and Eric L Schwartz. The isomap algorithm and topological stability. *Science*, 295(5552):7–7, 2002.

Chayan Banerjee, Kien Nguyen, Olivier Salvado, Truyen Tran, and Clinton Fookes. Pinns for medical image analysis: A survey. *arXiv*, 2024.

Achraf Ben-Hamadou, Nour Neifar, Ahmed Rekik, Oussama Smaoui, Firas Bouzguenda, Sergi Pujades, Edmond Boyer, and Edouard Ladroit. Teeth3Ds+: An Extended Benchmark for Intra-oral 3D Scans Analysis. *arXiv preprint arXiv:2210.06094*, 2022.

Ronny Bergmann and Daniel Tenbrinck. A graph framework for manifold-valued data. *SIAM Journal on Imaging Sciences*, 11(1):325–360, 2018.

Andrea L Bertozzi, Nadejda Drenska, Jonas Latz, and Matthew Thorpe. Partial differential equations in data science, 2025.

Karthik Bharath, Alexander Lewis, Akash Sharma, and Michael V Tretyakov. Sampling and estimation on manifolds using the langevin diffusion. *JMLR*, 2025.

Andrea Bonito and Ricardo Horacio Nochetto. *Geometric Partial Differential Equations*, volume 22. Elsevier, 2021.

Michele Giuliano Carlino and Elena Gaburro. Well balanced finite volume schemes for shallow water equations on manifolds. *Applied Mathematics and Computation*, 441:127676, 2023.

Woojin Cho, Minju Jo, Haksoo Lim, Kookjin Lee, Dongeun Lee, Sanghyun Hong, and Noseong Park. Parameterized physics-informed neural networks for parameterized PDEs. In *Forty-first International Conference on Machine Learning*, 2024.

Jae Choi, Yuzhou Chen, Huikyo Lee, Hyun Kim, and Yulia R Gel. Snn-pde: learning dynamic pdes from data with simplicial neural networks. In *AAAI*, 2024.

Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

Francisco Sahli Costabal, Simone Pezzuto, and Paris Perdikaris. δ-pinns: Physics-informed neural networks on complex geometries. *EAAI*, 2024.

Stéphane Cotin, Guillaume Mestdagh, and Yannick Privat. Organ registration from partial surface data in augmented surgery from an optimal control perspective. *Proceedings of the Royal Society A*, 480(2281):20230197, 2024.

Chinmay Datar, Taniya Kapoor, Abhishek Chandra, Qing Sun, Iryna Burak, Erik Lien Bolager, Anna Veselovska, Massimo Fornasier, and Felix Dietrich. Solving partial differential equations with sampled neural networks. *arXiv:2405.20836*, 2024.

Stefania Ebli, Michaël Defferrard, and Gard Spreemann. Simplicial neural networks. In *Topological Data Analysis and Beyond NeurIPS Workshop*, 2020.

L Rene I Fabregas and Jacob Rubinstein. A mathematical model for the progression of dental caries. *Mathematical Medicine and Biology: A Journal of the IMA*, 31(4):319–337, 2014.

Alfred Gierer and Hans Meinhardt. A theory of biological pattern formation. *Kybernetik*, 12(1): 30–39, 1972.

Faheem Gilani and John Harlim. Approximating solutions of linear elliptic pde's on a smooth manifold using local kernel. *Journal of Computational Physics*, 395:563–582, 2019.

Shira Faigenbaum Golovin and David Levin. Approximation of functions on manifolds in high dimension from noisy scattered data. *Jaen Journal on Approximation*, 13:43–73, 2022.

John Harlim, Shixiao Willing Jiang, and John Wilson Peoples. Radial basis approximation of tensor fields on manifolds: from operator estimation to manifold learning. *JMLR*, 2023.

Michael Holst and Martin Licht. Geometric transformation of finite element methods: Theory and applications. *Applied Numerical Mathematics*, 192:389–413, 2023.

Golriz Hosseinimanesh, Ammar Alsheghri, Julia Keren, Farida Cheriet, and Francois Guibault. Personalized dental crown design: A point-to-mesh completion network. *Medical Image Analysis*, 101:103439, 2025.

Wei-Fan Hu, Yi-Jun Shih, Te-Sheng Lin, and Ming-Chih Lai. A shallow physics-informed neural network for solving partial differential equations on static and evolving surfaces. *Computer Methods in Applied Mechanics and Engineering*, 418:116486, 2024.

Valerii Iakovlev, Markus Heinonen, and Harri Lähdesmäki. Learning continuous-time pdes from sparse data with graph neural networks. In *International Conference on Learning Representations*, 2021.

Prashant K Jha. From theory to application: A practical introduction to neural operators in scientific computing. *arXiv:2503.05598*, 2025.

Anran Jiao, Qile Yan, Jhn Harlim, and Lu Lu. Solving forward and inverse pde problems on unknown manifolds via physics-informed neural operators. *arXiv:2407.05477*, 2024.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

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

Rongjie Lai and Hongkai Zhao. Solving pdes on manifolds represented as point clouds and applications. In *Handbook of Numerical Analysis*, volume 20, pp. 315–349. 2019.

Guanhang Lei, Zhen Lei, Lei Shi, Chenyu Zeng, and Ding-Xuan Zhou. Solving pdes on spheres with physics-informed convolutional neural networks. *Applied and Computational Harmonic Analysis*, 74:101714, 2025.

P Li. Curvature and function theory on riemannian manifolds. *Surveys in Differential Geometry*, 7 (1):375–432, 2000.

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

Senwei Liang, Shixiao W Jiang, John Harlim, and Haizhao Yang. Solving pdes on unknown manifolds with machine learning. *Applied and Computational Harmonic Analysis*, 71:101652, 2024.

Yunfeng Liao, Jiawen Guan, and Xiucheng Li. Curvature-aware graph attention for pdes on manifolds. In *ICML*, 2025.

Lee Lindblom and Béla Szilágyi. Solving partial differential equations numerically on manifolds with arbitrary spatial topologies. *Journal of Computational Physics*, 243:151–175, 2013.

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

Marina Meilă and Hanyu Zhang. Manifold learning: What, how, and why. *Annual Review of Statistics and Its Application*, 11(1):393–417, 2024.

Masayasu Mimura, Hideo Sakaguchi, and Mitsugu Matsushita. Reaction–diffusion modelling of bacterial colony patterns. *Physica A: Statistical Mechanics and its Applications*, 282(1-2):283–303, 2000.

Reyhaneh Mir and Davoud Mirzaei. The d-rbf-pu method for solving surface pdes. *Journal of Computational Physics*, 479:112001, 2023.

James D. Murray. *Mathematical Biology I: An Introduction*, volume 17 of *Interdisciplinary Applied Mathematics*. Springer-Verlag New York, New York, 3rd edition, 2002. ISBN 978-0-387-95223-9. doi: 10.1007/b98868.

Luca Nieding, Cordula Reisch, Dirk Langemann, and Adrián Navas-Montilla. Impact of topography and combustion functions on fire front propagation in an advection-diffusion-reaction model for wildfires. *IFAC-PapersOnLine*, 59(1):103–108, 2025.

Theodore Papamarkou, Tolga Birdal, Michael Bronstein, et al. Position: Topological deep learning is the new frontier for relational learning. *PMLR*, 2024.

Mathilde Papillon, Sophia Sanborn, Johan Mathe, Louisa Cornelis, Abby Bertics, Domas Buracas, Hansen J Lillemark, Christian Shewmake, Fatih Dinc, Xavier Pennec, et al. Beyond euclid: An illustrated guide to modern machine learning with geometric, topological, and algebraic structures. *arXiv:2407.09468*, 2024.

Steven J Ruuth and Barry Merriman. A simple embedding method for solving partial differential equations on surfaces. *Journal of Computational Physics*, 227(3):1943–1961, 2008.

Junseung Ryu, Seungtae Park, and Hyung Ju Hwang. A physics-informed neural network framework for solving pdes on point clouds via surface reconstruction. *Neural Networks*, pp. 107928, 2025.

Ehsan Saleh, Saba Ghaffari, Timothy Bretl, Luke Olson, and Matthew West. Learning from integral losses in physics informed neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 43077–43111, 2024.

Hao Tang, Jiongyu Zhu, Zimeng Feng, Hao Li, and Chao Li. Geometric laplace neural operator. *arXiv preprint arXiv:2512.16409*, 2025.

Gabriel Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 902–907. IEEE, 1995.

Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

Greg Turk and Marc Levoy. Zippered polygon meshes from range images. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pp. 311–318, 1994.

Riccardo Valperga, Samuele Papa, and Stratis Gavves. M-pinns: Manifold-physics-informed neural networks for solving pdes on curved and changing domains. In *Greeks in AI Symposium 2025*, 2025.

Christopher Wang and Alex Townsend. Operator learning for hyperbolic partial differential equations. *JMLR*, 2025.

Haixin Wang, Yadi Cao, Zijie Huang, Yuxuan Liu, Peiyan Hu, Xiao Luo, Zezheng Song, Wanjia Zhao, Jilin Liu, Jinan Sun, et al. Recent advances on machine learning for computational fluid dynamics: A survey. *arXiv:2408.12171*, 2024.

Raymond O'Neil Wells and Oscar García-Prada. *Differential analysis on complex manifolds*, volume 21980. Springer New York, 1980.

Chenhui Xu, Dancheng Liu, Yuting Hu, Jiajie Li, Ruiyang Qin, Qingxiao Zheng, and Jinjun Xiong. Sub-sequential physics-informed learning with state space model. In *Forty-second International Conference on Machine Learning*, 2025.

Qile Yan, Shixiao W Jiang, and John Harlim. Kernel-based methods for solving time-dependent advection-diffusion equations on manifolds. *JSC*, 2023a.

Qile Yan, Shixiao Willing Jiang, and John Harlim. Spectral methods for solving elliptic pdes on unknown manifolds. *Journal of Computational Physics*, 486:112132, 2023b.

Shuai Yang, Ruikun Wang, Wenjie Zhao, and Yongzhen Ke. 3d intelligent scissors for dental mesh segmentation. *Computational and Mathematical Methods in Medicine*, 2020(1):1394231, 2020.

Bastian Zapf, Johannes Haubner, Miroslav Kuchta, Geir Ringstad, Per Kristian Eide, and Kent-Andre Mardal. Investigating molecular transport in the human brain from mri with physics-informed neural networks. *Scientific Reports*, 12(1):15475, 2022.

Yuval Zelig and Shai Dekel. Numerical methods for pdes over manifolds using spectral physics informed neural networks. *arXiv:2302.05322*, 2023.

Zhiyuan Zhao, Xueying Ding, and B Aditya Prakash. Pinnsformer: A transformer-based framework for physics-informed neural networks. In *ICLR*, 2024.

Dongzhe Zheng, Tao Zhong, and Christine Allen-Blanchette. Topology-preserving neural operator learning via hodge decomposition. *arXiv preprint arXiv:2605.13834*, 2026.

Youyi Zheng, Beijia Chen, Yuefan Shen, and Kaidi Shen. Teethgnn: semantic 3d teeth segmentation with graph neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 29(7): 3158–3168, 2022.

APPENDICES

A A. OVERVIEW OF PDE SOLVERS ON MANIFOLDS

Solving PDEs on unknown Riemannian manifolds is substantially more challenging and noticeably less investigated, both theoretically and numerically, than PDEs on Euclidean domains. In particular, PDEs defined on curved spaces where standard Euclidean geometry does not apply, necessitate specialized techniques to address complex geometry of the domain and finite data representations (Wells & García-Prada, 1980; Ruuth & Merriman, 2008; Bonito & Nochetto, 2021). This requires employing tools from differential geometry to define differential operators, designing computational frameworks for discretization on surfaces, and bringing in various embedding techniques such as closest point methodology, based on finding the closest representation of the surface and standard grid methods in certain embedding space. Solving PDEs on manifolds is also inherently connected with the machinery of manifold learning (Gilani & Harlim, 2019; Wang et al., 2024; Meilă & Zhang, 2024; Papillon et al., 2024; Bharath et al., 2025). The problem of solving PDEs on a manifold is aggravated even further in the case of the unknown manifold represented by a point cloud (Lai & Zhao, 2019). Some notable techniques include approximations of the non-Euclidean domain with a triangular mesh and then using various forms of the surface finite element method Lindblom & Szilágyi (2013); Holst & Licht (2023). However, constructing such triangular mesh is often infeasible when the observed data represent randomly sampled point clouds. The alternative is to use various collocation methods based on, for example, approximation via the global Radial Basis functions (Golovin & Levin, 2022; Yan et al., 2023b; Mir & Mirzaei, 2023; Harlim et al., 2023). However, when the unknown domain is identified only by point cloud data, such methods require a manifold parameterization scheme, which is computationally expensive and often results in the ill-conditioning of the differentiation matrices and associated limited convergence rates. More flexible approaches harness the graph theory and translating differential operators on manifolds into graph-based operations (Bergmann & Tenbrinck, 2018; Yan et al., 2023a). Despite being promising such methods are typically limited only to a particular type of differential operators. (For a more detailed discussion and comparison of methods see Jiao et al. (2024).)

B B. BACKGROUND

Riemannian Manifolds and Elements of Hodge Theory A Riemannian manifold $(\mathcal{M}, g)$ is a smooth manifold $\mathcal{M}$ equipped with a Riemannian metric tensor g, which assigns to each point $x \in \mathcal{M}$ a positive definite inner product g_x in the tangent space $T_x\mathcal{M}$. The cotangent space $T_p^*\mathcal{M}$ is the dual space of $T_p\mathcal{M}$, such that for any tangent vector $u \in T_p\mathcal{M}$ and cotangent vector $v^* \in T_p^*\mathcal{M}$, the duality pairing is given by: $v^*(u) := \langle v, u \rangle_{g_p}$. To better describe a manifold's structure on which the PDE is defined, we can leverage the tools of exterior calculus that extend the concepts of multivariable calculus such as the gradient and other differential operators to manifolds using differential forms.

Definition B.1. A differential p-form ω on $\mathcal{M}$ is a multilinear map $\omega_x : (T_x\mathcal{M})^p \to \mathbb{R}$, where $T_x^*\mathcal{M}$ is the dual space to the tangent space $T_p\mathcal{M}$. That is, ω_x can be viewed as a multilinear functional on p tangent vectors at each point of $\mathcal{M}$.

To study PDEs on manifolds, it is essential to relate the local differentiability and the global geometry of the domain, a connection provided by the Hodge theory. We start from defining an operator central to Hodge theory, namely, the Hodge star operator.

Definition B.2. Let $\mathcal{M}$ be a compact Riemannian manifold of dimension d. The *Hodge star operator* is an isomorphism $* : \Omega^p(M) \to \Omega^{d-p}(M)$, between the space of differential p-forms $\Omega^p(\mathcal{M})$ and $(d-p)$-forms $\Omega^{d-p}(\mathcal{M})$.

Equipped with the Hodge star operator, we proceed to define the inner product in p-forms.

Definition B.3. Let $\mathcal{M}$ be a compact Riemannian manifold and $\Omega^p(\mathcal{M})$ be a space of differential p-forms on $\mathcal{M}$ and $\alpha, \beta \in \Omega_c^p\mathcal{M})$. Then L^2 Hodge inner product on differential forms α and β is

$$(\alpha, \beta)_{L^2} = \int_{\mathcal{M}} \langle \alpha, \beta \rangle \, dV_g = \int_{\mathcal{M}} \alpha \wedge *\beta,$$

where $\langle \alpha, \beta \rangle$ is the pointwise inner product induced by the Riemannian metric g, $*$ is the Hodge star operator, and dV_g the Riemannian volume form.

The inner product on $\Omega^p(\mathcal{M})$ allows us now to define the analogues of differential operators on Euclidean domains to differential operators on p-forms, that is, we move from multivariable calculus on Euclidean domains to exterior calculus on Riemannian manifolds.

Definition B.4. The *exterior derivative* is a linear operator $d : \Omega^p(\mathcal{M}) \longrightarrow \Omega^{p+1}(\mathcal{M})$ and its adjoint codifferential operator $d^* : \Omega^{p+1}(\mathcal{M}) \to \Omega^p(\mathcal{M})$ such that $(d\alpha, \beta) = (\alpha, d^*\beta)$, where $\alpha \in \Omega^p(\mathcal{M})$ and $\beta \in \Omega^{p+1}(\mathcal{M})$. Finally, the *Hodge–Laplacian* on p-forms is a second-order differential operator $\Delta_H : \Omega^p(\mathcal{M}) \to \Omega^p(\mathcal{M})$ defined by $\Delta_H = dd^* + d^*d$.

In a nutshell, Hodge theory establishes a correspondence between Riemannnian manifold $\mathcal{M}$ and the analytic properties of differential operators on the space of p-forms.

B.1 Geometry Recovery from Point Clouds

Without loss of generality, let us consider a Riemannian manifold $\mathcal{M}$ of dimension 2. Given a point cloud $\mathcal{P} = \{x_i\}_{i=1}^N \subset \mathbb{R}^3$ sampled from an unknown two-dimensional Riemannian manifold $\mathcal{M}$, we estimate local intrinsic geometric quantities by reconstructing a polynomial approximation of a neighborhood of each sample point. The reconstructed local chart induces an approximate Riemannian metric, from which the Riemann curvature tensor is computed analytically. This procedure extracts intrinsic geometric information directly from discrete point samples without requiring a global parameterization of the underlying manifold.

Normal Vector Estimation For each point $x \in \mathcal{P}$, let $\mathcal{N}_k(x) = \{x_1, \dots, x_k\}$ denote its k-nearest neighbors. The neighborhood is centered at the origin, $\widetilde{x}_i = x_i - x, \qquad i = 1, \dots, k.$

For every pair $i < j$, we compute $\mathbf{n}_{ij} = \widetilde{x}_i \times \widetilde{x}_j$, which is orthogonal to the plane determined by the two neighboring vectors. Non-degenerate neighbor configurations are assumed so that sufficient independent tangent directions are available.

Whenever $\|\mathbf{n}_{ij}\| > \varepsilon$, the vector is normalized,

$$\widehat{\mathbf{n}}_{ij} = \frac{\mathbf{n}_{ij}}{\|\mathbf{n}_{ij}\|},$$

where throughout our experiments without loss of generality we set $\varepsilon = 5 \times 10^{-4}$.

A reference orientation is propagated locally through neighboring patches to resolve the sign ambiguity, where $\widehat{\mathbf{n}}_{\text{ref}}$ denotes the propagated reference orientation from neighboring patches. The final unit normal is then computed as

$$\mathbf{n}(x) = \frac{\sum_{i<j} \operatorname{sign}\left(\widehat{\mathbf{n}}_{ij} \cdot \widehat{\mathbf{n}}_{\text{ref}}\right) \widehat{\mathbf{n}}_{ij}}{\left\| \sum_{i<j} \operatorname{sign}\left(\widehat{\mathbf{n}}_{ij} \cdot \widehat{\mathbf{n}}_{\text{ref}}\right) \widehat{\mathbf{n}}_{ij} \right\|}.$$

Local Tangent Coordinates The tangent space at x is

$$T_x\mathcal{M} = \left\{ v \in \mathbb{R}^3 : v \cdot \mathbf{n}(x) = 0 \right\},$$

since $\mathcal{M}$ is a two-dimensional hypersurface embedded in $\mathbb{R}^3$.

A rigid rotation aligns $\mathbf{n}(x)$ with the positive z-axis. After rotation, every neighboring point is represented by local coordinates (u_i, v_i, w_i), where (u_i, v_i) are coordinates in the tangent plane and w_i is the signed distance to the tangent plane.

Surface Reconstruction and Metric Tensor Computation Assuming the manifold is locally smooth, we approximate it as the graph of a polynomial over the tangent plane,

$$w = h(u, v) = \sum_{m+n \le q} c_{mn} u^m v^n,$$

where q denotes the polynomial degree and the coefficients c_{mn} are estimated by weighted least-squares fitting over the local neighborhood. The resulting polynomial provides a smooth local

approximation of the manifold in a neighborhood of x. The reconstructed surface then defines the local parameterization

$$\mathbf{X}(u, v) = \big(u, v, h(u, v)\big)^T.$$

The induced Riemannian metric is

$$g_{\alpha\beta} = \left\langle \frac{\partial \mathbf{X}}{\partial x^\alpha}, \frac{\partial \mathbf{X}}{\partial x^\beta} \right\rangle, \qquad \alpha, \beta \in \{1, 2\},$$

where $x^1 = u$ and $x^2 = v$.

The tangent vectors are given by $\mathbf{X}_u = (1, 0, h_u)^T$, and $\mathbf{X}_v = (0, 1, h_v)^T$, which yield the metric tensor

$$g = \begin{pmatrix} 1 + h_u^2 & h_u h_v \\ h_u h_v & 1 + h_v^2 \end{pmatrix}.$$

Since $\det(g) = 1 + h_u^2 + h_v^2 > 0$, the inverse metric exists and is given by

$$g^{-1} = \frac{1}{\det(g)} \begin{pmatrix} g_{22} & -g_{12} \\ -g_{21} & g_{11} \end{pmatrix}.$$

The Christoffel symbols of the Levi–Civita connection are computed as

$$\Gamma^\alpha_{\beta\gamma} = \frac{1}{2} g^{\alpha\delta} \left(\frac{\partial g_{\delta\beta}}{\partial x^\gamma} + \frac{\partial g_{\delta\gamma}}{\partial x^\beta} - \frac{\partial g_{\beta\gamma}}{\partial x^\delta} \right),$$

where repeated indices are summed over $\delta \in \{1, 2\}$.

Since the metric components are algebraic functions of h and its derivatives, all required derivatives are computed symbolically, yielding closed-form expressions for the estimated Christoffel symbols.

Riemann Curvature Tensor The Riemann curvature tensor is computed as

$$R^\rho_{\sigma\mu\nu} = \frac{\partial \Gamma^\rho_{\nu\sigma}}{\partial x^\mu} - \frac{\partial \Gamma^\rho_{\mu\sigma}}{\partial x^\nu} + \Gamma^\rho_{\mu\lambda} \Gamma^\lambda_{\nu\sigma} - \Gamma^\rho_{\nu\lambda} \Gamma^\lambda_{\mu\sigma},$$

where repeated indices are summed over $\lambda \in \{1, 2\}$.

Finally, the symbolic expressions are evaluated at $(u, v) = (0, 0)$, corresponding to the original sample point x, yielding the estimated intrinsic Riemann curvature tensor at x.

Theoretical Guarantees of Manifold Recovery The reliability of the polynomial curvature estimation procedure is established through the approximation theory (Li, 2000).

Let $R(p)$ denote the true Riemann curvature tensor at a point $p \in \mathcal{M}$ and let $\bar{R}(\bar{p})$ denote the estimated tensor obtained from our procedure, where $\bar{p}$ is the projection of p onto the fitted polynomial surface. For any prescribed tolerance $\varepsilon > 0$, there exist a sampling resolution $\delta_0 > 0$ and a sufficient neighborhood size $N_0 \in \mathbb{N}$ such that

$$\|R(p) - \bar{R}(\bar{p})\|_\infty < \varepsilon,$$

whenever the point cloud spacing satisfies $\delta < \delta_0$ and the local neighborhood contains at least N_0 points. This guarantee follows from the approximation properties of polynomial interpolation, which ensure accurate recovery of the local manifold embedding and its derivatives required for curvature computation.

The following lemma establishes theoretical guarantees for the proposed interpolation-based approach to approximate the local geometry of an unknown manifold $\mathcal{M}$ from a point cloud $\mathcal{P}$ and provides an accuracy bound for curvature estimate.

Lemma B.5 (Approximation Bounds for Manifold Recovery)**.** *Given a smooth compact Riemannian manifold $\mathcal{M}$ embedded in $\mathbb{R}^3$ with induced metric g and a point $p \in \mathcal{M}$, assume that a point cloud $\mathcal{P} \subset \mathcal{M}$ admits a simplicial reconstruction K and mesh size $h = \max_{s \in K} \operatorname{diam}(s) < \infty$. Let R denote the Riemann curvature tensor of $\mathcal{M}$. Then, for any $\varepsilon > 0$, there exist a sampling resolution $\delta_h > 0$ and a sufficient number of samples N_h such that*

$$\|R(p) - \bar{R}(\bar{p})\|_\infty < \varepsilon,$$

where $\bar{R}$ is the curvature tensor computed on the interpolated surface $\bar{\mathcal{M}}$ obtained from N_h samples in a δ_h-neighborhood of p, and $\bar{p}$ is the projection of p onto $\bar{\mathcal{M}}$.

Proof. Let f denote the local embedding of $\mathcal{M}$ to $\mathbb{R}^3$. Then the metric components satisfy $g_{ij} = \partial_i f \cdot \partial_j f$, and the Riemann curvature tensor can be expressed in terms of derivatives of the metric, and therefore derivatives of f up to second order.

The interpolation theory (Li, 2000) guarantees the existence of a piecewise polynomial approximation P_h of degree k over neighboring simplices with mesh size h such that, for $f \in C^{k+1}(\mathcal{M})$,

$$\|f - P_h\|_\infty \leq C_0 h^{k+1},$$

where $C_0 = C_0(f, \mathcal{M})$ is a positive constant. Moreover, the errors satisfy

$$\|D^\alpha (f - P_h)\|_\infty \leq C h^{k+1-|\alpha|}$$

for $|\alpha| \leq k + 1$.

Since the curvature tensor depends on the embedding through derivatives up to 2nd order, the induced curvature approximation satisfies

$$\|R - \bar{R}\|_\infty \leq C_1 h^{k-1},$$

for a constant C_1 depending on $\mathcal{M}$ and the interpolation scheme. Hence, by selecting a sufficiently small mesh size h (equivalently, sufficiently dense local samples), the desired tolerance ε is achieved with $\delta = h$ and a corresponding number of samples □

C C. Additional Experimental Details

C.1 Shallow Water Systems

We discuss CASIN's advantages for complex PDEs defined on manifolds, using the shallow water system as our primary example. The shallow water system models the movement of a thin layer of water along a surface, tracking how it flows in response to gravity and the underlying terrain. In this settings, both the velocity and direction of flow are tightly coupled to the geometry of the surface that as water moves across a curved region, the curvature continuously redirects its path, bending the flow in ways that a flat-surface model would fail to capture. Curvature is the central factor governing the dynamics. The Shallow Water equations are given as follows:

Shallow Water System:

$$\partial_t h + \nabla_{\mathcal{M}} \cdot (h, \mathbf{u}) = 0 \tag{12}$$

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla_{\mathcal{M}})\mathbf{u} + \mathbf{u}^\perp + g \nabla_{\mathcal{M}} d = \mathbf{0} \tag{13}$$

where $\nabla_{\mathcal{M}}$ and $\nabla_{\mathcal{M}} \cdot$ are the intrinsic gradient and divergence on the manifold (Carlino & Gaburro, 2023).

CASIN's architecture offers a natural advantage for this system. Its simplicial encoding represents the surface using higher-order geometric structures, and flow is learned directly through these simplices. Because simplices inherently encode local geometric relationships between neighboring points, this design allows the local curvature of the surface to be passed directly into the model as it learns, rather than being approximated indirectly or inferred after the fact.

The empirical results reflect this advantage: CASIN achieves an error reduction of at least 20% over CurvGT across all three manifolds tested. This improvement also reduces in variance across runs, indicating that CASIN's gains are consistent rather than incidental. This is also confirmed by the one-sided t-test, indicating that the gains are statistically significant ($p < 0.001$).

C.2 Ground Truth Generation

For synthetic manifolds with known parameterizations (torus, distorted torus), we generate ground truth by solving each PDE on a dense regular grid (64×64) using finite differences with 4th order Runge-Kutta time integration, then interpolating solutions to the irregularly sampled point cloud via cubic interpolation. On the Stanford Bunny and dental meshes, lacking an analytic parameterization, we use the cotangent Laplacian on the mesh. For shallow water, we employ finite volume methods

Table 4: RMSE for the shallow-water system on synthetic manifolds (mean ± std). Best results in **bold**.

Model	Torus	Distorted Torus	Bunny
CurvGT	$0.0656_{\pm 2.3e\text{-}3}$	$0.0561_{\pm 6.5e\text{-}3}$	$0.0724_{\pm 1.2e\text{-}2}$
CASIN	$\mathbf{0.0466}_{\pm 9e\text{-}4}$	$\mathbf{0.0446}_{\pm 1.8e\text{-}3}$	$\mathbf{0.0506}_{\pm 7e\text{-}4}$

Table 5: Wall-clock and memory of CASIN vs. point-cloud size N (hidden=96, 4 layers). Parameters (336,866) and memory are essentially constant in N (with 5 runs).

N	$\lvert E\rvert$	$\lvert T\rvert$	**Del Triangulation**	**Fwd step**	**Rollout$_{40}$**	**Train step**
300	909	589	0.08 s	3.6 ms	0.14 s	52 ms
500	1,515	987	0.14 s	5.8 ms	0.22 s	62 ms
800	2,426	1,577	0.21 s	7.7 ms	0.30 s	85 ms
1200	3,623	2,370	0.31 s	10.7 ms	0.42 s	128 ms
1800	5,448	3,541	0.46 s	16.5 ms	0.64 s	172 ms

of hyperbolic systems. This ensures ground truth is computed independently of the graph-based discretization used by the neural network. On boundary conditions, since all synthetic manifolds (torus, distorted torus) are closed surfaces without boundary, no explicit boundary conditions are required and the topology is naturally handled by the simplicial complex construction. For dental meshes, tooth surfaces are similarly treated as closed manifolds after extraction from the full jaw scan, with the extraction boundary excluded from evaluation regions.

C.3 Data Description

Torus: We consider a standard torus embedded in $\mathbb{R}^3$ with major radius $R = 2.0$ and minor radius $r = 1.0$, parameterized via angular coordinates:

$$x(\theta, \phi) = (R + r\cos\phi)\cos\theta \tag{14}$$

$$y(\theta, \phi) = (R + r\cos\phi)\sin\theta \tag{15}$$

$$z(\theta, \phi) = r\sin\phi \tag{16}$$

C.4 Dental Data

We evaluate CASIN on the Teeth3DS dataset Ben-Hamadou et al. (2022), the first comprehensive public benchmark for intraoral 3D scan analysis, originally developed for the 3DTeethSeg Challenge at MICCAI 2022. The dataset comprises 1,800 high-resolution intraoral scans from 900 anonymized patients, with each patient contributing separate upper (maxillary) and lower (mandibular) jaw scans. In total, the dataset contains 23,999 individually annotated teeth with vertex-level labels following the FDI World Dental Federation notation system. While the dataset provides mesh, we **discard it** and recompute the See figure 3 for 4 samples of lower jaws of different patients.

C.5 Manifold Recovery Comparison

We also assess how well the curvature estimation recovers the underlying geometry from an unlabeled point cloud, against the true metric. CASIN's simplicial structure attains a median relative error of 4.9% , matching the curvature-aware CurvGT (4.5%) and outperforming classical baselines such as Isomap Balasubramanian & Schwartz (2002); Tenenbaum et al. (2000) (14.2%), which is unstable to the neighborhood size, and Diffusion Maps (Coifman & Lafon, 2006) (42.6%), whose diffusion distance is not an arc-length metric. The reconstruction also recovers the correct topology further validating that CASIN builds are topologically faithful.

Estimator	Recovery RMSE ↓	Height ↓	Edge-flow ↓
Taubin Shape Operator	0.015	0.024	0.0810
Cubic polynomial	0.086	0.027	0.0826
Quadratic polynomial	0.099	0.032	0.0805
Gauss-Bonnett	0.183	0.057	0.0820

Table 6: Sensitivity of CASIN performance to the curvature estimation method

C.6 Curvature Estimation Sensitivity of CASIN

We also provide a comparison of how different methods for curvature estimation impacts the performance of CASIN to delineate the contribution that CASIN's simplicial based architecture has in Table 6. We use four methods for curvature estimation, mainly a degree 2 polynomial fit, a degree 3 polynomial fit (as in the original framework), a mesh based method (Gauss-Bonnett) and Taubin Shape Operator Taubin (1995). We test it on torus for Shallow-Water equations 12. Our findings in 6 suggest that even though manifold recovery might vary by a lot across curvature estimation methods, CASIN's simplicial architecture allows it to be robust to noisy curvature estimates.

D D: Computational Details

Complexity. The cost of CASIN is governed by the size of the reconstructed simplicial complex, not by the point count N directly. Because the complex is built from a graph where each vertex has bounded degree, so the number of edges grows linearly with N. Each simplicial-convolution layer is a constant number of sparse matrix-vector products against the Hodge Laplacians L_0, L_1 (with $\mathcal{O}(|E|)$ non-zeros), giving a per-layer cost of $\mathcal{O}(N)$ and a total forward/backward cost of $\mathcal{O}(LN)$ for L layers. The parameter count is $\mathcal{O}(1)$ in N.

Empirical scaling. We verify this on the torus for $N \in \{300, \dots, 1800\}$. A log-log fit yields $|E| \sim N^{1.00}$ and per-training-step time $\sim N^{0.8}$ (the sub-linear exponent reflects fixed per-kernel cost amortizing over non-zeros; it will approach 1.0 asymptotically). The edge-to-vertex ratio is constant at $|E|/N \approx 3.0$ across the range, confirming linear growth. Wall-clock is reported in Table 5.

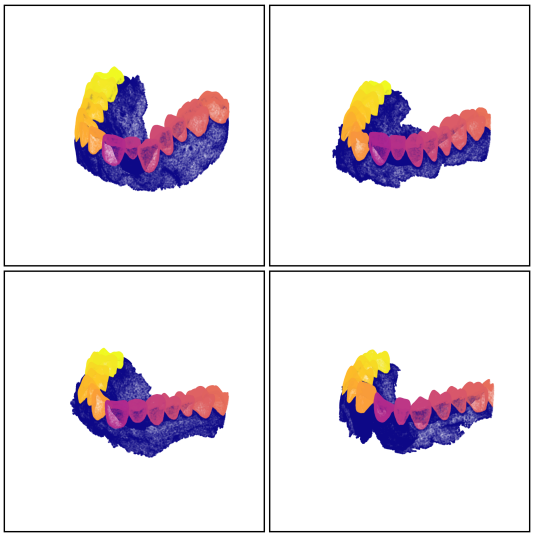


Figure 3: Examples from the Teeth3DS dataset.

D.1 Reproducibility Details

We provide all the necessary scripts, code, data and seeds to reproduce results in the code files submitted along with this appendix. CASIN uses 4 simplicial convolution layers of hidden dimension 96, trained with Adam (learning rate 3×10^{-3}, cosine annealing, gradient clipping at 1.0) for 500 epochs on 800 sampled points. Results are averaged over 5 random seeds. All the training was done on an Nvidia RTX 4080 GPU.