Predicting Long-term Dynamics of Complex Networks via Identifying Skeleton in Hyperbolic Space

Ruikun Li
Shenzhen International Graduate
School, Tsinghua University
Shenzhen, China

Huandong Wang*
Department of Electronic Engineering
BNRist, Tsinghua University
Beijing, China

Jinghua Piao
Department of Electronic Engineering
BNRist, Tsinghua University
Beijing, China

Qingmin Liao
Shenzhen International Graduate
School, Tsinghua University
Shenzhen, China

Yong Li
Department of Electronic Engineering
BNRist, Tsinghua University
Beijing, China

ABSTRACT

Learning complex network dynamics is fundamental for understanding, modeling, and controlling real-world complex systems. Though great efforts have been made to predict the future states of nodes on networks, the capability of capturing long-term dynamics remains largely limited. This is because they overlook the fact that long-term dynamics in complex network are predominantly governed by their inherent low-dimensional manifolds, i.e., skeletons. Therefore, we propose the Dynamics-Invariant Skeleton Neural Network (DiskNet), which identifies skeletons of complex networks based on the renormalization group structure in hyperbolic space to preserve both topological and dynamics properties. Specifically, we first condense complex networks with various dynamics into simple skeletons through physics-informed hyperbolic embeddings. Further, we design graph neural ordinary differential equations to capture the condensed dynamics on the skeletons. Finally, we recover the skeleton networks and dynamics to the original ones using a degree-based super-resolution module. Extensive experiments across three representative dynamics as well as five real-world and two synthetic networks demonstrate the superior performances of the proposed DiskNet, which outperforms the state-of-the-art baselines by an average of 10.18% in terms of long-term prediction accuracy. Code for reproduction is available at: https://github.com/tsinghua-fib-lab/DiskNet.

CCS CONCEPTS

• **Computing methodologies** → **Modeling methodologies**; *Learning latent representations*; **Machine learning**; • **Applied computing** → *Physics*; *Physics*; • **Networks** → **Network dynamics**.

KEYWORDS

Complex Network, Dynamical Systems, Graph Neural Networks, Neural ODE

1 INTRODUCTION

The evolution behavior of numerous complex systems in the real world, such as the brain [9], social networks [51], and ecosystems [15], can be modeled as the dynamics on complex networks [4] where the components within the system are regarded as nodes in the network, and the coupling interactions between components are regarded as edges [13]. Learning the dynamics of these complex networks are instrumental to the ability to analyze them, facilitating numerous important applications including understanding their inherent resilience [13], predicting their future states [31, 35], and controlling their states [28], etc. However, a vast number of nodes and edges typically exist in complex networks. On the one hand, the large-scale network significantly increases the computational complexities of inference and estimation tasks on complex networks. On the other hand, the large number of less important nodes, acting as interference variables, potentially prevent us from uncovering the effective dynamics of the network [13]. In this context, a fundamental question arises: given an arbitrary complex network, can we find a lower-dimensional skeleton of it, on which the coarse-scale long-term dynamics can be modeled with both high prediction performance and low computation cost?

As a long-standing problem, numerous methods have been developed to reduce the dimensionality of large-scale networks, including statistical physics methods [41] and machine learning methods [20, 22]. In particular, the statistical physics methods utilize the renormalization group technique [14, 40] to identify the skeleton that retains the degree distribution and clustering coefficient. The machine learning methods [20, 22] learn graph coarsening strategies that preserve the topological and node feature consistency between the original network and the network that is reconstructed using the coarsened network. All these methods only preserve the static topological and nodal information, which ignores the dynamics on the network [14, 20, 22, 40]. Consequently, they fail to capture the collective behavior of nodal dynamics to

accurately forecast their long-term evolution. Additionally, several analytical methods have shown that the dynamics of complex networks in terms of key properties, e.g., resilience [13] or critical time delays [29], can be embedded into a submanifold with ultra-low dimensionality [13, 29]. However, they rely on overly simplified assumptions regarding network topology, e.g., homogeneous node degrees or degree-based mean-field [29, 39], and cannot obtain a skeleton with a desired dimensionality. Taken together, whether the complex networks and their intricate dynamics can be condensed into low-dimensional skeletons is still an open problem.

Nevertheless, developing an effective approach to learn the skeleton and the corresponding dynamics on the skeleton network is also a difficult task with the following challenges. First, identifying the skeleton that facilitates the most accurate predictions of the evolutionary behavior of complex networks is challenging. The skeleton can be obtained by coarsening a cluster of nodes into a super-node. However, we lack sufficient understanding in terms of how to find such clusters while preserving the intrinsic dynamics in the obtained skeleton network. Second, how to establish a reversible mapping between the node states in the skeleton network and the original network is the second challenge. It involves establishing an aggregating function to calculate the state of the super-node based on its sub-nodes, and a lifting function to map the state of each super-node back to its sub-nodes. Since the aggregation process will introduce inevitably information loss, how to build an efficient lifting function is a crucial problem.

In this paper, we propose a novel deep learning framework, named Dynamics-Invariant Skeleton Neural Network (DiskNet), to identify the low-dimensional skeleton of complex networks in hyperbolic space for modeling their long-term dynamics. This framework identifies the skeleton through physics-informed embeddings. These embeddings are initialized based on renormalization group techniques developed in hyperbolic space and then fine-tuned in the end-to-end training process, enabling the combination of knowledge and data to overcome the first challenge. Then, DiskNet employs a graph neural ordinary differential equation to efficiently model the condensed dynamics on the skeleton network. Finally, it utilizes an elaborately-designed degree-based super-resolution module as the lifting function to map the condensed dynamics back to the original network. This module utilizes the homogeneity of nodes with respect to their degrees to identify clusters of nodes that share the same lifting network, thereby constructing an effective lifting function to address the second challenge.

Our contribution can be summarized as follows:

- We propose to identify the skeleton based on hyperbolic embeddings inspired by physical knowledge from the renormalization group theory, enabling us to combine knowledge and data to learn an effective skeleton.
- We develop a powerful graph neural ordinary differential equations (ODEs) integrated with a novel super-resolution module utilizing the homogeneity of nodes with close degrees, allowing us to accurately model the condensed dynamics on the skeleton and map it to the original network.
- Extensive experimental results on three representative network dynamics on five real-world and two synthetic network topologies show that DiskNet outperforms state-of-the-art

baselines by an average of 10.18% in terms of prediction accuracy, indicating its superiority.

2 PROBLEM FORMULATION

Without loss of generality, the network topology is represented by an adjacency matrix $A \in \{0, 1\}^{N \times N}$, where each element $a_{ij} = 1$ indicates the connection between nodes i and j. Combining it with the node set $\mathcal{V}$, a complex network can be modeled as a graph $G = (\mathcal{V}, A)$. The network dynamics describe the evolution of node states on the graph G, taking into account the self-dynamics of each node and the interactions with its neighbors, following the equation

$$\frac{dx_i}{dt} = f(x_i) + \sum_{j \neq i}^{N} a_{ij} g(x_i, x_j), \quad (1)$$

where f represents the self-dynamics and g denotes the coupling dynamics between nodes. The node state matrix, denoted as $\mathbf{X} \in \mathbb{R}^{N \times L \times d}$, represents the observed values of the d-dimensional states of N nodes over a continuous sequence of L time steps.

Furthermore, we consider the skeleton of the graph G as $G_s = (\mathcal{V}_s, A_s)$, where $A_s \in \{0, 1\}^{\gamma N \times \gamma N}$ describes the connection between the super-nodes, which are defined as the aggregation of original nodes, in the set $\mathbb{V}_s$. Here, γ represents the reduction ratio of the skeleton relative to the original topology, given by $\gamma = \frac{|\mathcal{V}_s|}{|\mathcal{V}|}$. The correspondence between the super-nodes and the original nodes is defined by the assignment matrix $P \in \{0, 1\}^{\gamma N \times N}$, where $p_{ij} = 1$ indicates that node j is aggregated into super-node i. The function $x_{s,i} = u(x_{P_i})$ describes how the state of super-node i is aggregated from its sub-nodes. The dynamics of the network on the skeleton, denoted as $\frac{dx_{s,i}}{dt}$, is defined in the same form as Eqn. 1. The state of a super-node is mapped back to its sub-nodes as $x_i = v(x_{s,i})$ by the lifting function v. We emphasize that obtaining the network skeleton through aggregation rather than pruning is aimed at consolidating the collective dynamics of similar nodes into a representative super-node.

Our goal is to determine the assignment matrix P, state aggregation function u, and lifting function v to establish the low-dimensional skeleton G_s of the graph G. Subsequently, by modeling the dynamics $\frac{dx_{s,i}}{dt}$ on the skeleton and mapping it back to the sub-nodes, we aim to achieve an accurate prediction of the long-term evolution of node states $X_{horizon}$, given history observations $X_{lookback}$. The whole process is illustrated in Figure 1.

3 METHOD

In this section, we propose Dynamics-Invariant Skeleton Neural Network (DiskNet) to predict the long-term dynamics of complex networks via identifying skeleton in hyperbolic space. The overall framework is illustrated in Figure 2. According to the challenges mentioned above, we first propose a hyperbolic renormalization group (RG) module which leverages the powerful representation capability of hyperbolic geometry to capture the topological and dynamic similarities of nodes and adaptive calculates the assignment matrix, guiding the node assignment and dynamics aggregation. Then, a neural ODE model is designed to model the neural dynamics of super-nodes on the skeleton, incorporating two mechanisms: self-dynamics and neighbor interactions. Finally, a degree-based
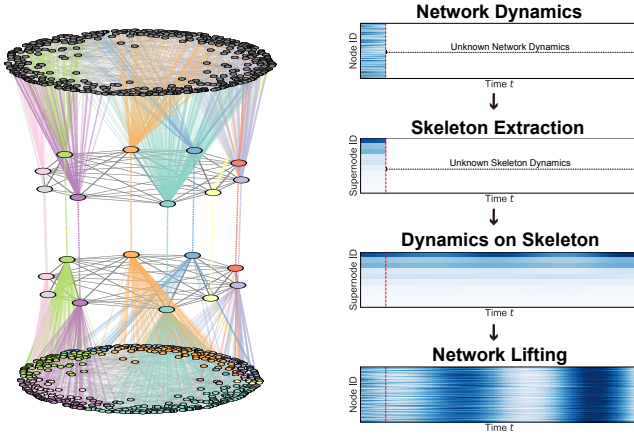
Figure 1: Long-term prediction via the skeleton of complex network dynamics.

clustering super-resolution module is proposed to lift the dynamic representation on the skeleton to the original nodes.

3.1 Hyperbolic Renormalization Group

Given the adjacency matrix A of a complex network, the key to identifying the skeleton A_s of the network lies in determining the assignment matrix P based on the topological properties of the network and the latent similarity of node dynamics. We work in the hyperbolic space [26] to measure the similarity between the original nodes and the super-nodes. Based on this, we derive an adaptive assignment matrix to aggregate the dynamic representations of the sub-nodes for each super-node, which joint incorporates features of the topology and dynamics. We introduce prior physical knowledge by initializing the hyperbolic embeddings of the super-nodes to guide model training.

3.1.1 Learnable hyperbolic embeddings. The Poincaré disk is a 2-dimensional hyperbolic geometry where the negative curvature property aligns the hyperbolic distances between node embeddings with their proximity in the graph. Hyperbolic spaces of the same dimension have a larger capacity than Euclidean spaces, which makes it more suitable for representing both the topological and dynamic information of nodes in large-scale networks. The distance between two points on the Poincaré disk is given by

$$d_H^c(x, y) = \frac{1}{\sqrt{|c|}} \text{arcosh}(1 - \frac{2c||x - y||^2}{(1 + c||x||^2)(1 + c||y||^2)}). \quad (2)$$

Previous works have focused on learning hyperbolic embeddings $C^H \in \mathbb{R}^{N \times 2}$ for network topology, which is frozen in DiskNet after initialization and used for hierarchical representation of the original topology. Our core idea is to maintain a learnable hyperbolic embedding $C_s^H \in \mathbb{R}^{\gamma N \times 2}$ for each super-node to capture the dynamic characteristics of all its sub-nodes. We first initialize C_s^H according to the method described in Sec. 4.3.4 to introduce prior physical knowledge, and then fine-tune it during end-to-end training. Considering that the computational rules defined in Euclidean space do not apply to vectors in hyperbolic space, we project C^H and C_s^H onto corresponding Euclidean coordinates using the logarithmic map

$$\theta_z^E = \frac{2}{\sqrt{|c|}\lambda_z^c} \text{arctanh}(\sqrt{|c|}|| - z \oplus_c \theta^H||) \frac{-z \oplus_c \theta^H}{|| - z \oplus_c \theta^H||}, \quad (3)$$

where $\oplus_c$ means Möbius addition [30] and $\lambda_x^c = \frac{2}{1 + c||x||^2}$. θ^H and θ^E denote the vector in hyperbolic and Euclidean space, respectively. Consistent with most of the work [26, 32, 34, 42], the tangent space chosen for the hyperbolic manifold is at the origin point $\mathbb{O}$, and the curvature is set as -1. We provide detailed verification of the superiority of hyperbolic embeddings for representing both the topological and dynamic similarities of nodes in Sec. 4.3.2.

3.1.2 Adaptive assignment matrix. Based on the hyperbolic embeddings of the original nodes and super-nodes, we compute the adaptive assignment matrix as

$$P = \text{softmax}(\tilde{C}_s \tilde{C}^T), \quad (4)$$

where softmax is calculated row-wise, $\tilde{C} = MLP(C^E)$ and $\tilde{C}_s = MLP(C_s^E)$. We minimize $L_E = \frac{1}{N} \sum_{i=1}^{N} H(P_i)$, where H denotes the entropy function and P_i is the i-th column of P, to constrain each column of the assignment matrix to be close to a one-hot vector. Additionally, we minimize $L_R = ||A, P^T P||_F$ to heuristically guide the assignment matrix in preserving the skeleton of the original topology, where $|| \cdot ||_F$ denotes the Frobenius norm.

3.1.3 Node dynamics aggregation. We employ graph convolutional neural networks to capture the propagation of dynamics in the original nodes and aggregate them into dynamic representations of the super-nodes as

$$H = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X)\Theta_1) \in \mathbb{R}^{N \times d},$$
$$X_s = PH \in \mathbb{R}^{\gamma N \times d}, \quad (5)$$

where $\tilde{D} = \sum_j \tilde{A}_{ij}$, $\tilde{A} = A + I$ and Θ is learnable parameters. The connections between super-nodes are described by the adjacency matrix $A_s = PAP^T$, where an element $a_{s,ij}$ indicates the total number of connections between sub-nodes within a super-node ($i = j$) or between different super-nodes ($i \neq j$).

3.1.4 Physics-informed initialization. Statistical physics assigns the meaning in terms of node similarity to angular coordinates of the node embeddings in the Poincaré disk and employs the renormalization group based on angular coordinates [14]. This design helps preserve the degree distribution and clustering coefficient of the network topology in the skeleton. While this method does not guarantee an effective representation of network dynamics, DiskNet can automatically adjust the learnable hyperbolic embeddings of super-nodes during subsequent end-to-end training. Therefore, we sort the original nodes according to angular coordinates and divide them into groups of $\frac{1}{\gamma}$ nodes to predefine the assignment matrix P_0, which is used for pretraining the adaptive assignment module with the loss function of $L_p = \frac{1}{\gamma N^2} \sum |P - P_0|$. We provide the pseudo-code of the pre-training phase in the appendix C. The initial values of hyperbolic embeddings for super-nodes are set as $c_{s,i}^H = \frac{1}{|P_{0,i}|} \sum_{j \in P_{0,i}} c_j^H$, where $P_{0,i}$ denotes the set of sub-nodes belonging to the super-node i. We extensively analyze the relationship between angular coordinates and node dynamics in Sec. 4.3.2.
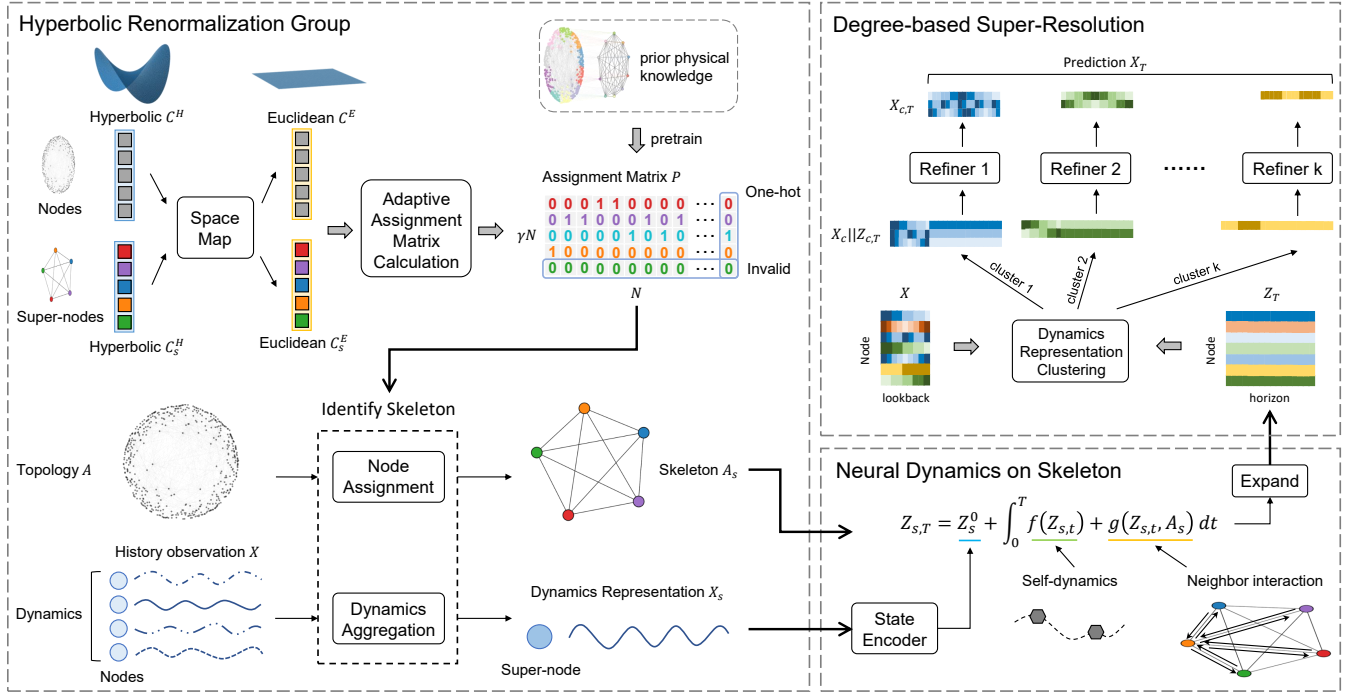
Figure 2: Overall framework of DiskNet: (1) Hyperbolic Renormalization Group, which identifies the representation and skeleton of network dynamics; (2) Neural Dynamics on Skeleton, which models the dynamics of super-nodes on the skeleton; and (3) Degree-based Super-Resolution, which lifts the predicted values of super-nodes to the original nodes.

3.2 Neural Dynamics on Skeleton

After obtaining the skeleton of network dynamics and the aggregated state of super-nodes, we maintain the form of network dynamics as Eqn. 1 and model it using the neural ODE [8].

3.2.1 ODE function. We now define the forward ODE function for the latent dynamics $\frac{dZ_s}{dt}$ of the skeleton. Taking into account that the evolution of each super-node $x_{s,i}$ is influenced by its self-dynamics and coupling dynamics, the parameterized time derivative consists of two terms: one donating the self-dynamics function $f(Z_s)$ and another representing the interaction with neighbors $g(Z_s, A_s)$. Thus the dynamic equation for each super-node is given by

$$\frac{dZ_s}{dt} = f(Z_s) + g(Z_s, A_s), \tag{6}$$

where f is an MLP and g is a GNN responsible for information propagation

$$g(Z_s, A_s) = \sigma(A\sigma(Z_s\Theta_3)\Theta_2). \tag{7}$$

3.2.2 Solve. Given the ODE function, the predicted trajectory of the skeleton dynamics can be solved by any ODE solver as an initial value problem

$$Z_{s,T} = Z_{s,0} + \int_0^T f(Z_{s,t}) + g(Z_{s,t}, A_s)dt, \tag{8}$$

where the initial state $Z_{s,0} = MLP(X_s) \in \mathbb{R}^{\gamma N \times 1}$. This allows us to predict the state of super-nodes at any continuous time point T.

3.3 Lift to Original Nodes

After obtaining the predicted trajectories of super-nodes on the skeleton, it is necessary to lift them back to each individual node in the original graph to complete the prediction of network dynamics. However, there is inevitably information loss during the aggregation of super-nodes, such as the heterogeneity of sub-node states within the same super-node. Simply copying the predicted values of super-nodes to their sub-nodes is not sufficient for accurate expanding. To overcome this challenge, inspired by research in statistical physics [13, 29], we designed a super-resolution module based on degree clustering.

3.3.1 Degree-based clustering. By employing a degree-weighted mean-field approach, one can compress the state and tipping point of degree-similar nodes into a super-node for prediction [13, 27, 29]. This implies that the collective dynamics of nodes exhibit homogeneity with respect to their degrees. Inspired by this, considering that real-world networks often follow power-law distributions, we choose to use the logarithm of degrees as features for clustering nodes by K-means algorithm and perform super-resolution within each cluster. In fact, in the Poincaré disk, the degree of a node directly corresponds to its radial coordinate. Therefore, nodes with similar radial coordinates should be identified as belonging to the same super-node.

3.3.2 Super-resolution. Based on the effective representation of the skeleton dynamics, the design of the super-resolution module does not need to be complex. We denote the set of nodes in the c-th cluster as N_c, with their historical observations as X_c and

predicted values of the corresponding super-node as $Z_{c,T}$. Since the expand operation simply copies the predicted values of the super-node to its sub-nodes without differentiation, $Z_{c,T}$ is a coarse and homogeneous prediction for each node. We refine $Z_{c,T}$ based on the historical observations X_c, resulting in accurate prediction as

$$X_{c,T} = \sigma((h_0||h_1)\Theta_4), \tag{9}$$

where $||$ donates the concatenation operation, $h_0 = MLP(X_c)$ and $h_1 = MLP(Z_{c,T})$. Thus, we have completed the prediction of network dynamics. We demonstrate in Sec. 4.4.2 that the prediction accuracy rapidly reaches the upper limit as the cluster count k increases. Therefore, a limited setting of k is sufficient to achieve good prediction performance.

3.4 Training

We now describe the training process of DiskNet. Before the start of end-to-end training, we first initialize the hyperbolic embeddings of both the original nodes and super-nodes, as described in Sec. 4.3.4, and pretrain the adaptive assignment module leveraging the prior physical knowledge. Subsequently, in each iteration, the model learns the adaptive assignment matrix guided by the loss functions L_E and L_R. The prediction of the skeleton dynamics is updated using the L2 error as $L_s = MSE(\overline{Z}_{s,T}, Z_{s,T})$, where $\overline{Z}_{s,T}$ represents the predicted states of the super-node, and $Z_{s,T}$ is the ground truth obtained based on the assignment matrix and the state encoder. Finally, the L2 error of the prediction for all original nodes, $L_p = MSE(\overline{X}_T, X_T)$, is utilized for updating the super-resolution module. Due to the Poincaré disk being a conformal map of Euclidean space, when updating node embeddings with gradients, it is necessary to scale the gradients according to the Riemannian metric tensor

$$\nabla_H(\theta) = (\frac{1 - ||\theta||}{2})^2 \nabla_E(\theta), \tag{10}$$

where $\nabla_E(\theta)$ and $\nabla_H(\theta)$ is the Euclidean and hyperbolic gradient of vector θ, respectively. After completing end-to-end training, we separately fine-tune the super-resolution module to enhance the prediction performance.

4 EXPERIMENTS

In this section, we present the evaluation results of DiskNet. We first describe the dynamics and network topology used for testing, followed by an introduction to the baseline for comparison. Then, we analyze the long-term prediction performance, interpretability of the identified skeleton, and sensitivity analysis. Detailed information about the software and hardware environment for all experiments is provided in the appendix B.

4.1 Experiment Setup

We conduct experiments on three representative nonlinear dynamic systems, operating on five real-world network topologies and two synthetic topologies. The details of dataset generation can be found in the appendix A.2.

4.1.1 Network Topology. We consider five real-world network topologies [36] from infrastructure, social networks, brain networks, and

the World Wide Web [1]: (a) power grids of the United States (*PowerGrid*); (b) brain networks of *Drosophila*; (c) mutually liked Facebook pages (*Social*); (d) pages linking to www.epa.gov (*Web*); (e) ties between two non-US-based airports (*Airport*). Additionally, we also consider two synthetic networks: (f) Barabási–Albert network (*BA*) [3]; and (g) Watts–Strogatz network (*WS*) [43]. The statistical characteristics of all topologies are shown in Table 1.

4.1.2 Network Dynamics. We considered three representative dynamics from biology and physics: (a) Hindmarsh-Rose dynamics [6], (b) FitzHugh-Nagumo dynamics [12], and (c) Coupled Rössler dynamics [1]. The governing equations and details of parameter settings are provided in the appendix A.1.

4.2 Baseline

For all the datasets, we compare with the following GNN-based state-of-the-art methods.

- **DCRNN** [24] employs bidirectional random walks and an encoder-decoder architecture with scheduled sampling to enable accurate spatio-temporal forecasting.
- **GraphWaveNet** [44] incorporates an adaptive dependency matrix that captures hidden spatial dependencies through node embeddings.
- **AGCRN** [2]: introduces two adaptive modules along with recurrent networks to automatically capture fine-grained spatial and temporal correlations in traffic series.
- **NDCN** [50] captures continuous-time dynamics on complex networks by combining neural ODEs and GNNs.
- **STGNCDE** [10] combines spatio-temporal graph neural networks with neural controlled differential equations for predicting multi-variate time series.
- **MTGODE** [19] abstracts multivariate time series into dynamic graphs with time-evolving node features and models their continuous dynamics in the latent space.
- **FourierGNN** [48] is the state-of-the-art GNN model for multi-variate time series forecasting. It introduces hypervariate graphs and performs matrix multiplications in Fourier space.

4.3 Performance Evaluation

In this section, we analyze the long-term prediction performance, identified skeletons, and computational cost of DiskNet.

4.3.1 Long-term Prediction. We evaluated the performance of DiskNet in long-term predictions for all topologies and dynamics based on Mean Absolute Error (MAE). The model was provided with observed trajectories from the past 12 time steps and tasked with predicting the future 120 steps of node states. In all experiments, the reduction ratio γ and cluster count k of DiskNet is set to 50% and 10, respectively, and the reasons for this setting are explained in Sec. 4.4.

The average performance for 120 time steps are shown in Table 1. It is noteworthy that DiskNet consistently outperforms all baselines by a large margin in long-term prediction performance across almost all scenarios, with the improvement reaching over 50% in some cases. Furthermore, it can be observed that the suboptimal

---
[1]https://networkrepository.com/index.php

Table 1: Average performance of the long-term prediction with 120 time steps with standard deviation from 10 runs. The best results are highlighted in bold, and the suboptimal results are emphasized with an underline.

		PowerGrid	Drosophila	Social	Web	Airport	BA	WS
	Nodes	5,300	1,770	3,892	4,252	2,904	5,000	5,000
	Edges	8,271	8,905	17,239	8,896	15,644	14,991	10,000
	Avg degree	3.121	10.062	8.859	4.184	10.774	5.996	4.000
	Avg clustering coeff	0.088	0.265	0.374	0.071	0.456	0.010	0.368
	MAE ↓	*Mean ± Std.*	*Mean ± Std.*	*Mean ± Std.*	*Mean ± Std.*	*Mean ± Std.*	*Mean ± Std.*	*Mean ± Std.*
Hindmarsh-Rose	DCRNN	0.3652 ± 0.0315	0.5616 ± 0.0392	0.4869 ± 0.0478	0.4226 ± 0.0453	0.5457 ± 0.0434	0.4284 ± 0.0552	0.3436 ± 0.0330
	GraphWaveNet	0.6083 ± 0.0080	0.6300 ± 0.0011	0.6144 ± 0.0012	0.6124 ± 0.0039	0.6448 ± 0.0064	0.6148 ± 0.0057	0.6081 ± 0.0027
	AGCRN	0.2417 ± 0.0093	0.3590 ± 0.0078	0.3236 ± 0.0053	0.2801 ± 0.0091	0.3591 ± 0.0064	0.2964 ± 0.0072	0.2422 ± 0.0114
	NCDN	0.1554 ± 0.0027	0.3255 ± 0.0031	0.2938 ± 0.0101	0.2298 ± 0.0092	0.3349 ± 0.0030	0.2672 ± 0.0082	0.2002 ± 0.0065
	STGNCDE	0.3283 ± 0.0173	0.4836 ± 0.0161	0.4171 ± 0.0111	0.3817 ± 0.0227	0.4802 ± 0.0081	0.3952 ± 0.0157	0.3379 ± 0.0118
	MTGODE	**0.1283 ± 0.0045**	<u>0.3170 ± 0.0141</u>	<u>0.2264 ± 0.0057</u>	<u>0.1791 ± 0.0083</u>	<u>0.3131 ± 0.0085</u>	<u>0.1895 ± 0.0075</u>	**0.1295 ± 0.0057**
	FourierGNN	0.1651 ± 0.0053	0.3793 ± 0.0119	0.3024 ± 0.0083	0.2378 ± 0.0112	0.3830 ± 0.0110	0.2251 ± 0.0198	0.1601 ± 0.0073
	DiskNet	<u>0.1290 ± 0.0029</u>	**0.2515 ± 0.0124**	**0.2086 ± 0.0052**	**0.1645 ± 0.0018**	**0.2769 ± 0.0028**	**0.1878 ± 0.0027**	<u>0.1359 ± 0.0031</u>
FitzHugh-Nagumo	DCRNN	0.4487 ± 0.0368	0.5136 ± 0.0859	0.4780 ± 0.0993	0.4810 ± 0.0599	0.5297 ± 0.0659	0.4809 ± 0.1139	0.3597 ± 0.0170
	GraphWaveNet	0.5805 ± 0.0060	0.5036 ± 0.0013	0.5523 ± 0.0006	0.4538 ± 0.0016	0.5127 ± 0.0011	0.5223 ± 0.0026	0.5904 ± 0.0007
	AGCRN	0.2903 ± 0.0264	0.2100 ± 0.0175	0.2272 ± 0.0105	0.1924 ± 0.0070	0.2136 ± 0.0142	0.2130 ± 0.0122	0.2179 ± 0.0132
	NCDN	0.4281 ± 0.0048	0.2750 ± 0.0058	0.3550 ± 0.0056	0.2467 ± 0.0092	0.2965 ± 0.0046	0.3193 ± 0.0074	0.4291 ± 0.0101
	STGNCDE	0.3165 ± 0.0229	0.2645 ± 0.0215	0.2534 ± 0.0168	0.2918 ± 0.0336	0.2633 ± 0.0243	0.2532 ± 0.0127	0.2573 ± 0.0048
	MTGODE	0.1420 ± 0.0133	0.1331 ± 0.0138	**0.0826 ± 0.0054**	<u>0.0873 ± 0.0034</u>	<u>0.0934 ± 0.0052</u>	0.0906 ± 0.0039	0.1103 ± 0.0037
	FourierGNN	<u>0.1359 ± 0.0106</u>	<u>0.1301 ± 0.0100</u>	0.1315 ± 0.0116	0.1016 ± 0.0054	0.1392 ± 0.0076	0.1171 ± 0.0168	<u>0.0677 ± 0.0091</u>
	DiskNet	**0.0997 ± 0.0076**	**0.0851 ± 0.0021**	<u>0.0932 ± 0.0061</u>	**0.0575 ± 0.0024**	**0.0917 ± 0.0029**	**0.0800 ± 0.0037**	**0.0591 ± 0.0072**
Coupled Rössler	DCRNN	0.5825 ± 0.0277	0.4927 ± 0.0235	0.4719 ± 0.0340	0.4062 ± 0.0139	0.3977 ± 0.0415	0.5046 ± 0.0417	0.4655 ± 0.0173
	GraphWaveNet	0.4394 ± 0.0496	0.3743 ± 0.0329	0.3665 ± 0.0093	0.2969 ± 0.0528	0.3465 ± 0.0088	0.3951 ± 0.0136	0.3555 ± 0.0113
	AGCRN	0.1837 ± 0.0146	0.1562 ± 0.0065	0.1563 ± 0.0059	0.1188 ± 0.0030	0.1603 ± 0.0065	0.1699 ± 0.0060	0.1387 ± 0.0073
	NCDN	0.5908 ± 0.0145	0.5077 ± 0.0235	0.4992 ± 0.0219	0.3743 ± 0.0130	0.4567 ± 0.0133	0.4593 ± 0.0297	0.5104 ± 0.0177
	STGNCDE	0.2621 ± 0.0595	0.2021 ± 0.0362	0.2487 ± 0.0626	0.1877 ± 0.0337	0.3004 ± 0.0641	0.2891 ± 0.0535	0.1727 ± 0.0316
	MTGODE	<u>0.0707 ± 0.0075</u>	0.0896 ± 0.0125	**0.0907 ± 0.0076**	0.0753 ± 0.0125	**0.0824 ± 0.0104**	<u>0.0748 ± 0.0085</u>	<u>0.0860 ± 0.0044</u>
	FourierGNN	0.1013 ± 0.0067	<u>0.0849 ± 0.0060</u>	0.1368 ± 0.0063	<u>0.0689 ± 0.0021</u>	0.1070 ± 0.0040	0.0905 ± 0.0094	0.1281 ± 0.0078
	DiskNet	**0.0612 ± 0.0042**	**0.0640 ± 0.0032**	<u>0.0878 ± 0.0148</u>	**0.0680 ± 0.0035**	<u>0.0909 ± 0.0111</u>	**0.0616 ± 0.0116**	**0.0761 ± 0.0089**

model varies across different scenarios, even among state-of-the-art models, indicating fluctuations in their performance. This, on the one hand, demonstrates that our approach is capable of accurately identifying and modeling the skeleton of network dynamics in a wide range of scenarios. On the other hand, it emphasizes the significance of the low-dimensional skeleton for stable and accurate long-term predictions. In contrast, the NCDN and MTGODE model, which also employ graph neural ODE, perform well only in a few scenarios, possibly because it models the entire network without effectively aggregating the crucial collective behaviors of nodes.

We present the detailed statistics of the percentage improvement of DiskNet compared to the suboptimal baseline across all topologies for different dynamics in Figure 3, which demonstrates the performance of DiskNet in terms of prediction error at different time steps. In terms of short-term prediction performance, DiskNet is slightly weaker than the suboptimal baseline due to inevitable information loss during the aggregation of node states. However, this short-term information gradually diminishes as the time scale increases, and DiskNet excels in capturing long-term evolution through the skeleton.

4.3.2 Dynamics Skeleton. We then study the skeleton of the dynamic identified by DiskNet through comparative experiments and visual analysis. We compare five baselines including heuristics, graph pooling and renormalization group: (a) random assignment; (b) threshold assignment based on node degree; (c) threshold assignment based on betweenness centrality; (d) DiffPool [49]; (e) GMPool
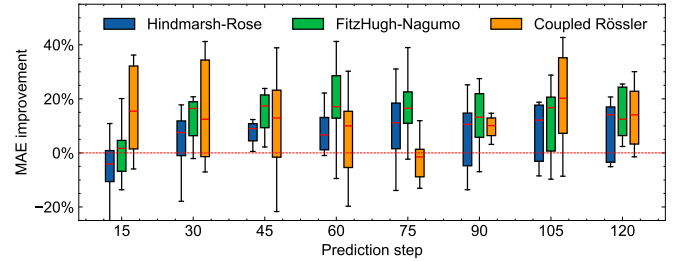
Figure 3: Percentage improvement in MAE as a function of the horizon for different dynamics.

[21]; and (f) static RG [14]. The first three methods select super-nodes according to certain rules, and the sub-nodes are assigned randomly to super-nodes. The assignment rules of two graph pooling methods are consistent with the original paper. The RG method assigns nodes solely based on their topological features, thus it is referred to as static RG. In all experiments, to ensure fairness, after obtaining the assignment matrix using the baseline, modules such as state aggregation and skeleton dynamics, are trained end-to-end using DiskNet. The reduction ratio γ was set to 50% for all models.

We conduct experiments on Hindmarsh-Rose dynamics across all topologies as shown in Table 2. With the powerful representation capability of hyperbolic space, the skeleton identified by DiskNet captures the long-term dynamics of the nodes, resulting in more accurate predictions compared to the baseline in all scenarios.

(a) RG (no dynamics)   (b) DiskNet (Hindmarsh-Rose)   (c) DiskNet (FitzHugh-Nagumo)   (d) DiskNet (Coupled Rössler)
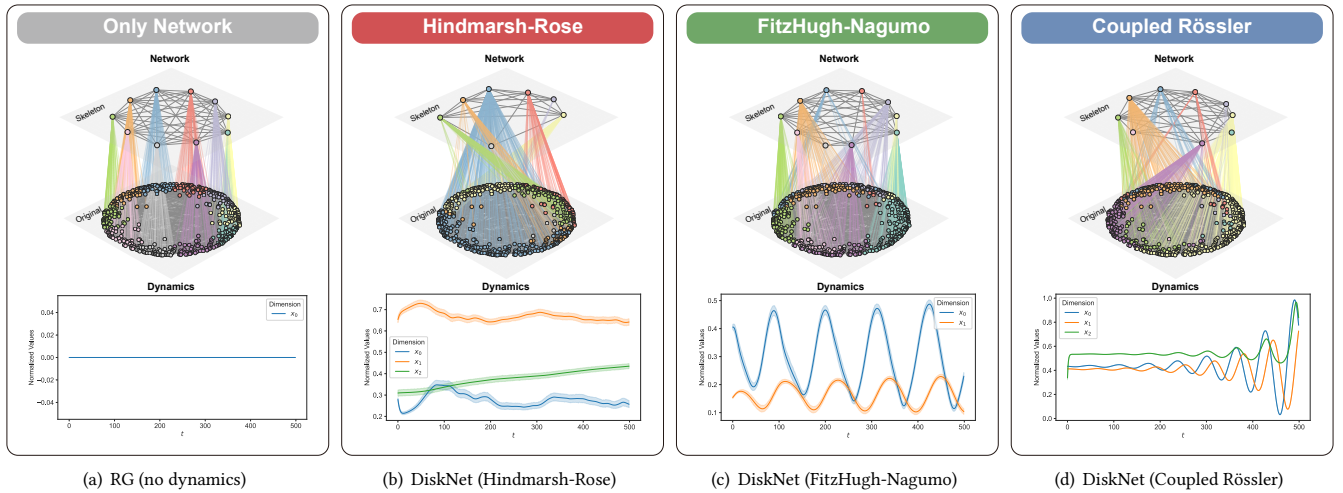
Figure 4: Identified skeleton for different dynamics on the BA-500 networks of DiskNet and static RG model. The bottom half of each figure depicts the average evolution trajectory of all nodes.

Table 2: Comparison of the prediction MAE with 120 time steps for Hindmarsh-Rose dynamics among different skeleton identification methods.

	PowerGrid	Drosophila	Social	Web	Airport	BA	WS
Random	0.1813	0.2891	0.2497	0.1986	0.2842	0.2169	0.1748
Degree	0.1817	0.2895	0.2437	0.1997	0.2902	0.2187	0.1874
Betweenness	0.1802	0.2948	0.2485	0.1997	0.2897	0.2197	0.1567
DiffPool	0.1757	0.2863	0.2376	0.1804	0.2785	0.2147	0.1815
GMPool	0.1698	0.2821	0.2299	0.1756	0.2873	0.2069	0.1618
static RG	0.1517	0.2640	0.2179	0.1763	0.2796	0.1909	0.1572
DiskNet	**0.1290**	**0.2515**	**0.2086**	**0.1645**	**0.2769**	**0.1878**	**0.1359**

Heuristic methods, by solely considering the topological characteristics while disregarding the node dynamics, exhibit inferior performance compared to random assignment in certain scenarios. Furthermore, graph pooling methods, which represent node dynamics in Euclidean space to derive the assignment matrix, are unable to effectively measure node similarity and capture latent dynamic interactions as effectively as hyperbolic space. Similar observations have also been seen in node classification tasks [32, 33]. Although the static RG method also utilizes hyperbolic geometry to represent node features, it only considers static topological features for calculating the assignment matrix, disregarding the similarity in node dynamics. As a result, it is unable to identify the skeleton of long-term dynamics.

We visualize the skeletons identified by the static RG method and DiskNet for the same topology but different dynamics in Figure 4, where the dynamic curves represent the average trajectories of all nodes. We utilize the result of a 500-node BA network with reduction ratio $\gamma = 0.02\%$ to avoid visual clutter caused by large network size (visualized skeletons for real-world networks in Table 1 are provided in the appendix D.1). Since the static RG method does not consider dynamics, the identified skeletons are the same for the three dynamics, as shown in Figure 4(a). In contrast, DiskNet shows variations in the identified skeletons for different dynamics. For FitzHugh-Nagumo dynamics (Figure 4(c)) and Coupled Rössler

dynamics (Figure 4(d)), where all nodes exhibit oscillatory behavior, DiskNet identifies a similar number of super-nodes, and the assignment patterns are also similar. However, for the Hindmarsh-Rose dynamics, which have heterogeneous node dynamics but a simple trend, the corresponding skeleton tends to be captured with fewer super-nodes.

Further analysis of the skeletons identified by DiskNet reveals that the majority of super-nodes share the same local neighborhood with their sub-nodes, but there are also instances where the positions of sub-nodes deviate, especially in the case of Hindmarsh-Rose dynamics (Figure 4(b)). While the existing work [14] has demonstrated that nodes with similar angles on the Poincaré disk should belong to the same super-node when considering only topological characteristics, DiskNet suggests that this conclusion no longer holds when considering node dynamics. Even nodes with substantially different angular coordinates may be assigned to the same super-node due to similar dynamic properties. Furthermore, it is observed that nodes with the same angular coordinates may be assigned to different super-nodes due to differences in radial coordinates, which is attributed to the correlation between node degrees and their dynamics [29]. Finally, it is worth noting that some super-nodes are only assigned a small number of sub-nodes, which is further explained in Sec. 4.4.1.

4.3.3 Computational Cost. DiskNet only requires integration of the forward ODE function on the skeleton during inference, which significantly saves computational resources compared to previous neural ODE-based methods. We compare DiskNet with varying reduction ratio γ and two other neural ODE-based baselines on different topologies to measure the time cost for single-sample inference. The results in Figure 5 demonstrate that even with the additional computational overhead from the super-resolution module, DiskNet with $\gamma = 50\%$ achieves performance comparable to NDCN. In fact, as the network size increases, the dimension reduction potential of network dynamics often exceeds 50% (which we validate in Sec. 4.4.1). Therefore, DiskNet has higher practical value
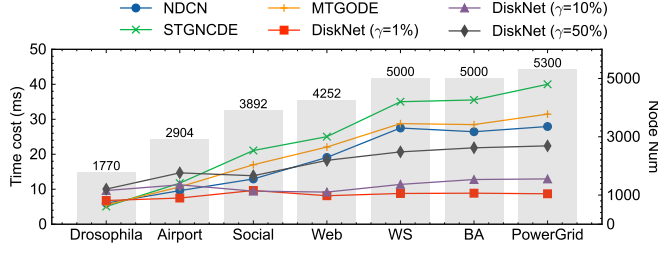
Figure 5: Time cost per iteration on different network topologies for Hindmarsh-Rose dynamics.

Table 3: Comparison of the prediction MAE with 120 time steps for Coupled Rössler dynamics among different versions of DiskNet. We do not initialize the hyperbolic embedding and pre-training, denoted as *w/o Emb* and *w/o PT*.

	PowerGrid	Drosophila	Social	Web	Airport	BA	WS
DiskNet	**0.0612**	**0.0640**	**0.0878**	**0.0680**	**0.0909**	**0.0616**	**0.0761**
w/o Emb	0.1166	0.1014	0.1269	0.0864	0.1172	0.1118	0.1262
w/o PT	0.1297	0.1018	0.1299	0.0879	0.1160	0.0911	0.0814
w/o Emb & PT	0.1377	0.1080	0.1020	0.0910	0.1215	0.1149	0.0838

than previous neural ODE-based methods in predicting large-scale network dynamics.

4.3.4 Physics-informed Initialization. The performance of DiskNet in Table 1 are based on the physics-informed initialization, which allows the model to start learning the optimal parameters from a good initial point. We evaluate versions of DiskNet on the Coupled Rössler dynamics that (1) do not initialize the hyperbolic embeddings of super-nodes and/or (2) do not pre-train the assignment matrix calculation module. The results (in Table 3) show that while DiskNet without physical initialization performs competitively compared to the baseline in many scenarios, it is always inferior to the physically initialized version, confirming the necessity of physics-informed initialization.

4.4 Sensitivity Analysis

In this section, we analyze the impact of two important hyperparameters, namely the reduction ratio γ and the clustering count k, on the prediction performance of DiskNet.

4.4.1 Reduction ratio γ. The reduction ratio γ sets an upper limit on the number of super-nodes in the skeleton. Although the model automatically determines the assignment relationship based on the topological and dynamics properties, allowing some super-nodes unassigned to any sub-nodes, the proportion of super-nodes does not exceed γ. We conducted tests on the Coupled Rössler dynamics in both the BA network and PowerGrid network, as shown in Figure 6. We define the occupancy ratio as the proportion of super-nodes that are assigned at least one sub-node out of the total available super-nodes. It is observed that when γ is less than 1%, the prediction performance is the worst, and the occupancy ratio is the highest. This indicates that the upper limit of super-nodes is too low to identify an appropriate skeleton. After γ exceeds 25%, the prediction performance becomes similar. This suggests both the low-dimensional characteristics of network dynamics and
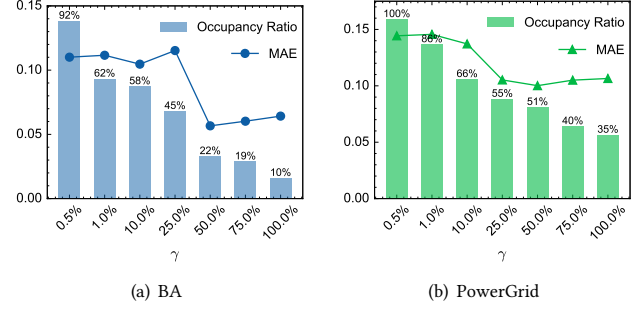

(a) BA

(b) PowerGrid

Figure 6: MAE and Occupancy ratio as functions of γ for Coupled Rössler dynamics on PowerGrid and BA networks.

(a) Social MSE

(b) Web MSE
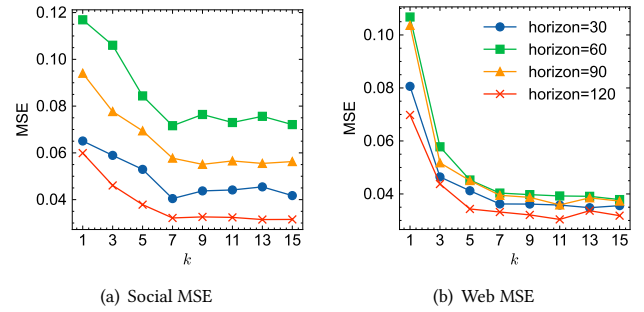
Figure 7: MSE of different horizons as a functions of k for FitzHugh-Nagumo dynamics on Social and Web networks.

indicates that setting γ to 50% is generally adequate, as the model is capable of adaptively determining the appropriate occupancy ratio. Finally, although both networks consist of approximately 5,000 nodes, the occupancy ratio of the PowerGrid network is slightly higher than that of the BA network for the same γ. This suggests that the Coupled Rössler dynamics in the PowerGrid network has relatively higher dimensions and pose greater prediction challenges, which corresponds to the prediction errors in Table 1.

4.4.2 Cluster count k. The cluster count k represents the number of refiners in the super-resolution module and is positively correlated with the prediction performance. However, it needs to be balanced with the parameter size of the model. We conduct prediction tests at different time scales (horizon) on the FitzHugh-Nagumo dynamics in both the Social network and Web network, which is shown in Figure 7. When k is less than 7, increasing k leads to lower prediction errors. However, beyond k of 7, further increasing k does not significantly improve the performance. Therefore, we recommend using the default value of 10 for k in general, which balances both performance and computational cost.

5 RELATED WORK

5.1 Learning Network Dynamics

The dynamics of many real-world systems, characterized by non-linearity and multi-scale properties [23], are often abstracted into complex network models, thereby illustrating the interactions between nodes [5, 13, 25]. With the development of deep learning

techniques such as GNNs, data-driven modeling of complex network dynamics has garnered significant attention [11]. Murphy et al. [31] proposed a GNN architecture that accurately models disease propagation on networks with minimal assumptions about the dynamics. NCDN [50] combined neural ODE [8] and GNNs to model continuous-time dynamics of complex networks for the first time. Huang et al. [16, 17] successfully modeled dynamic topology and cross-environment network dynamics by introducing the edge dynamics ODE and environment encoders, respectively. However, despite the demonstration [35, 38] that network dynamics evolve in a remarkably low-dimensional subspace, existing approaches still model the evolution of dynamics over the entire topology, which fails to capture the low-dimensional structures that dominate the long-term evolution of the network.

5.2 Dimension Reduction of Complex Network

Driven by the high costs of storage and computational expenses in large-scale networks, Serrano et al. [37] proposed the disparity filter to ensure the retention of small-scale components during the pruning process of multiscale networks. García-Pérez et al. [14] designed the hyperbolic renormalization group method based on the assumption of angular similarity to identify the low-dimensional skeleton of complex networks and Villegas et al. [40] proposed a Laplacian renormalization group difusion-based picture for heterogeneous networks. Kumar et al. [22] used node feature consistency as guidance and learned the low-dimensional skeleton through end-to-end training. Additionally, some works on motif learning [18, 45] treat each motif in the original graph as a super-node, thus constructing a skeleton that aggregates higher-order geometric characteristics. However, these methods only consider the static properties of networks and cannot guide the modeling of network dynamics. Gao et al. [13] captured macroscopic dynamics by mapping the original system to an effectively representative one-dimensional system and further developed a degree-based dimensionality reduction method [29] that successfully captures the critical point of phase transitions in heterogeneous networks. However, these traditional expert knowledge-based methods cannot automatically identify the most suitable skeleton based on the collective behavior exhibited of nodes for different topologies and dynamics.

5.3 Hyperbolic Graph Learning

The negative curvature property of hyperbolic geometry leads to nonlinear distance metrics, enabling it to better capture the hierarchical relationships and nonlinear dependencies in graph-structured data. Nickel et al. [32, 33] embedded symbolic data into the Poincaré ball and Lorentz model, capturing the hierarchical structure and similarity of unstructured data. Liu et al. [26] and Chami et al. [7] introduced the hyperbolic geometry into graph representation learning through Riemannian manifold mapping, significantly improving the performance of node classification tasks. Yang et al. [46, 47] further utilized hyperbolic geometry to model the hierarchical structure of time-varying networks. Although these works have demonstrated the suitability of hyperbolic geometry for representing network structure data, there is still a lack of its application in modeling complex network dynamics.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we explore the low-dimensional skeleton of complex network dynamics and its crucial role in long-term prediction through a skeleton identification approach based on hyperbolic node embeddings. By representing the collective dynamics behavior of nodes in the hyperbolic space, we compute the adaptive assignment matrix to identify the skeleton. Then, we achieve effective long-term prediction by modeling the dynamics of super-nodes and lifting them to the original nodes. Extensive experimental results demonstrate that our model outperforms the baseline with a significant margin in terms of accuracy and robustness. In addition, we validate the superior capability of hyperbolic geometry over Euclidean space in representing the low-dimensional structure of network dynamics. We demonstrate the limitations of solely identifying skeletons based on topological characteristics through visualization results, which showcase distinct skeletons resulting from different dynamics on the same topology. The analysis results indicate that, while ensuring prediction accuracy, there is rich potential for dimensionality reduction in the dynamics of different types of large-scale networks. In the future, we plan to explore how the topology and dynamics of complex networks affect the maximum achievable dimensionality reduction while ensuring prediction accuracy. Additionally, we aim to design a mechanism for automatically selecting the reduction ratio γ.

ACKNOWLEDGMENTS

REFERENCES

[1] Alex Arenas, Albert Díaz-Guilera, Jurgen Kurths, Yamir Moreno, and Changsong Zhou. 2008. Synchronization in complex networks. *Physics reports* 469, 3 (2008), 93–153.
[2] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. 2020. Adaptive graph convolutional recurrent network for traffic forecasting. *Advances in neural information processing systems* 33 (2020), 17804–17815.
[3] Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. *science* 286, 5439 (1999), 509–512.
[4] Baruch Barzel and Albert-László Barabási. 2013. Universality in network dynamics. *Nature physics* 9, 10 (2013), 673–681.
[5] Baruch Barzel, Yang-Yu Liu, and Albert-László Barabási. 2015. Constructing minimal models for complex system dynamics. *Nature communications* 6, 1 (2015), 7186.
[6] FS Borges, Ewandson L Lameu, Kelly C Iarosz, Paulo R Protachevicz, Iberê L Caldas, Ricardo L Viana, Elbert EN Macau, Antonio M Batista, and Murilo da Silva Baptista. 2018. Inference of topology and the nature of synapses, and the flow of information in neuronal networks. *Physical Review E* 97, 2 (2018), 022303.
[7] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. 2019. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems* 32 (2019).
[8] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. 2018. Neural ordinary differential equations. *Advances in neural information processing systems* 31 (2018).
[9] Dante R Chialvo. 2010. Emergent complex neural dynamics. *Nature physics* 6, 10 (2010), 744–750.
[10] Jeongwhan Choi, Hwangyong Choi, Jeehyun Hwang, and Noseong Park. 2022. Graph neural controlled differential equations for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 6367–6374.
[11] Jingtao Ding, Chang Liu, Yu Zheng, Yunke Zhang, Zihan Yu, Ruikun Li, Hongyi Chen, Jinghua Piao, Huandong Wang, Jiazhen Liu, et al. 2024. Artificial Intelligence for Complex Network: Potential, Methodology and Application. *arXiv preprint arXiv:2402.16887* (2024).
[12] Richard FitzHugh. 1961. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical journal* 1, 6 (1961), 445–466.
[13] Jianxi Gao, Baruch Barzel, and Albert-László Barabási. 2016. Universal resilience patterns in complex networks. *Nature* 530, 7590 (2016), 307–312.

[14] Guillermo García-Pérez, Marián Boguñá, and M Ángeles Serrano. 2018. Multiscale unfolding of real networks by geometric renormalization. *Nature Physics* 14, 6 (2018), 583–589.

[15] Jiliang Hu, Daniel R Amor, Matthieu Barbier, Guy Bunin, and Jeff Gore. 2022. Emergent phases of ecological diversity and dynamics mapped in microcosms. *Science* 378, 6615 (2022), 85–89.

[16] Zijie Huang, Yizhou Sun, and Wei Wang. 2021. Coupled graph ode for learning interacting system dynamics. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 705–715.

[17] Zijie Huang, Yizhou Sun, and Wei Wang. 2023. Generalizing graph ode for learning complex system dynamics across environments. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 798–809.

[18] Muhammad Ifte Khairul Islam, Max Khanov, and Esra Akbas. 2023. MPool: motif-based graph pooling. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 105–117.

[19] Ming Jin, Yu Zheng, Yuan-Fang Li, Siheng Chen, Bin Yang, and Shirui Pan. 2022. Multivariate time series forecasting with dynamic graph neural odes. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[20] Yu Jin, Andreas Loukas, and Joseph JaJa. 2020. Graph coarsening with preserved spectral properties. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 4452–4462.

[21] Sung Moon Ko, Sungjun Cho, Dae-Woong Jeong, Sehui Han, Moontae Lee, and Honglak Lee. 2023. Grouping matrix based graph pooling with adaptive number of clusters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 8334–8342.

[22] Manoj Kumar, Anurag Sharma, Shashwat Saxena, and Sandeep Kumar. 2023. Featured graph coarsening with similarity guarantees. In *International Conference on Machine Learning*. PMLR, 17953–17975.

[23] Ruikun Li, Huandong Wang, and Yong Li. 2023. Learning slow and fast system dynamics via automatic separation of time scales. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4380–4390.

[24] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[25] Chang Liu, Jingtao Ding, Yiwen Song, and Yong Li. 2024. TDNetGen: Empowering Complex Network Resilience Prediction with Generative Augmentation of Topology and Dynamics. *to appear in Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2024).

[26] Qi Liu, Maximilian Nickel, and Douwe Kiela. 2019. Hyperbolic graph neural networks. *Advances in neural information processing systems* 32 (2019).

[27] Xueming Liu, Daqing Li, Manqing Ma, Boleslaw K Szymanski, H Eugene Stanley, and Jianxi Gao. 2022. Network resilience. *Physics Reports* 971 (2022), 1–108.

[28] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. 2011. Controllability of complex networks. *nature* 473, 7346 (2011), 167–173.

[29] Cheng Ma, Gyorgy Korniss, Boleslaw K Szymanski, and Jianxi Gao. 2023. Generalized dimension reduction approach for heterogeneous networked systems with time-delay. *arXiv preprint arXiv:2308.11666* (2023).

[30] Emile Mathieu, Charline Le Lan, Chris J Maddison, Ryota Tomioka, and Yee Whye Teh. 2019. Continuous hierarchical representations with poincaré variational auto-encoders. *Advances in neural information processing systems* 32 (2019).

[31] Charles Murphy, Edward Laurence, and Antoine Allard. 2021. Deep learning of contagion dynamics on complex networks. *Nature Communications* 12, 1 (2021), 4720.

[32] Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems* 30 (2017).

[33] Maximillian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International conference on machine learning*. PMLR, 3779–3788.

[34] Wei Peng, Jingang Shi, Zhaoqiang Xia, and Guoying Zhao. 2020. Mix dimension in poincaré geometry for 3d skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*. 1432–1440.

[35] Bastian Prasse and Piet Van Mieghem. 2022. Predicting network dynamics without requiring the knowledge of the interaction graph. *Proceedings of the National Academy of Sciences* 119, 44 (2022), e2205517110.

[36] Ryan Rossi and Nesreen Ahmed. 2015. The network data repository with interactive graph analytics and visualization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.

[37] M Ángeles Serrano, Marián Boguná, and Alessandro Vespignani. 2009. Extracting the multiscale backbone of complex weighted networks. *Proceedings of the national academy of sciences* 106, 16 (2009), 6483–6488.

[38] Vincent Thibeault, Antoine Allard, and Patrick Desrosiers. 2024. The low-rank hypothesis of complex systems. *Nature Physics* (2024), 1–9.

[39] Chengyi Tu, Paolo D'Odorico, and Samir Suweis. 2021. Dimensionality reduction of complex dynamical systems. *Iscience* 24, 1 (2021).

[40] Pablo Villegas, Tommaso Gili, Guido Caldarelli, and Andrea Gabrielli. 2023. Laplacian renormalization group for heterogeneous networks. *Nature Physics* 19, 3 (2023), 445–450.

[41] Huandong Wang, Huan Yan, Can Rong, Yuan Yuan, Fenyu Jiang, Zhenyu Han, Hongjie Sui, Depeng Jin, and Yong Li. 2023. Multi-Scale Simulation of Complex Systems: A Perspective of Integrating Knowledge and Data. *Comput. Surveys* (2023).

[42] Xiao Wang, Yiding Zhang, and Chuan Shi. 2019. Hyperbolic heterogeneous information network embedding. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5337–5344.

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

[44] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121* (2019).

[45] Rong Yan, Peng Bao, Huawei Shen, and Xuanya Li. 2023. Learning node representation via Motif Coarsening. *Knowledge-Based Systems* 278 (2023), 110821.

[46] Menglin Yang, Min Zhou, Marcus Kalander, Zengfeng Huang, and Irwin King. 2021. Discrete-time temporal network embedding via implicit hierarchical learning in hyperbolic space. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1975–1985.

[47] Menglin Yang, Min Zhou, Hui Xiong, and Irwin King. 2022. Hyperbolic temporal network embedding. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[48] Kun Yi, Qi Zhang, Wei Fan, Hui He, Liang Hu, Pengyang Wang, Ning An, Longbing Cao, and Zhendong Niu. 2023. FourierGNN: Rethinking Multivariate Time Series Forecasting from a Pure Graph Perspective. *arXiv preprint arXiv:2311.06190* (2023).

[49] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems* 31 (2018).

[50] Chengxi Zang and Fei Wang. 2020. Neural dynamics on complex networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 892–902.

[51] Jorge Gomez Tejeda Zañudo, Gang Yang, and Réka Albert. 2017. Structure-based control of complex networks with nonlinear dynamics. *Proceedings of the National Academy of Sciences* 114, 28 (2017), 7234–7239.

A DATASETS

A.1 Network Dynamics

Hindmarsh-Rose dynamics [6] simulates the spiking activity of neurons in the brain during information processing, with the following governing equations

$$\begin{cases} \dfrac{dx_{i,1}}{dt} = x_{i,2} - ax_{i,1}^3 + bx_{i,1}^2 - x_{i,3} + I_{ext} + \epsilon(V_{syn} - x_{i,1}) \sum_{j=1}^{N} A_{ij}\mu(x_{j,1}), \\[2mm] \dfrac{dx_{i,2}}{dt} = c - ux_{i,1}^2 - x_{i,2}, \\[2mm] \dfrac{dx_{i,3}}{dt} = r[s(x_{i,1} - x_0) - x_{i,3}], \end{cases} \tag{11}$$

where parameters $a = 1$, $b = 3$, $c = 1$, $u = 5$, $s = 4$, $r = 0.005$, $x_0 = -1.6$, coupling strength $\epsilon = 0.15$, $V_{syn} = 2$, $\lambda = 10$, $\Omega_{syn} = 1$, and external current is $I_{ext} = 3.24$ for all neurons. The coupling term is

$$\mu(x_{j,1}) = \frac{1}{1 + e^{[-\lambda(x_{j,1} - \Omega_{syn})]}}. \tag{12}$$

FitzHugh-Nagumo dynamics [12] is mainly used for describing the activation and deactivation dynamics of spiking neurons. The equations governing the dynamics are

$$\begin{cases} \dfrac{dx_{i,1}}{dt} = x_{i,1} - x_{i,1}^3 - x_{i,2} - \epsilon \sum_{j=1}^{N} A_{ij} \dfrac{x_{j,1} - x_{i,1}}{k_i^{in}}, \\[2mm] \dfrac{dx_{i,2}}{dt} = a + bx_{i,1} + cx_{i,2}, \end{cases} \tag{13}$$

where parameters $\epsilon = 1$, $a = 0.28$, $b = 0.5$, $c = -0.04$, and k_i^{in} is the in-degree of neuron i.

Coupled Rössler dynamics [1] describes the dynamics of n oscillators with natural frequencies following a normal distribution, in which the self-dynamics are heterogeneous

$$\begin{cases} \dfrac{dx_{i,1}}{dt} = -w_i x_{i,2} - x_{i,3} + \epsilon \displaystyle\sum_{j=1}^{N} A_{ij}(x_{j,1} - x_{i,1}), \\[2mm] \dfrac{dx_{i,2}}{dt} = w_i x_{i,1} + a x_{i,2}, \\[2mm] \dfrac{dx_{i,3}}{dt} = b + x_{i,3}(x_{i,1} + c), \end{cases} \tag{14}$$

where parameters $\epsilon = 0.15$, $a = 0.2$, $b = 0.2$, $c = -6$, and frequencies $w \sim \mathcal{N}(1, \sigma)$ with $\sigma = 0.1$.

A.2 Simulation

We employ Euler's method to numerically solve the dynamics of the aforementioned systems for each topology, generating evolutionary trajectories as the dataset with a time step of 0.01 seconds. The Hindmarsh-Rose model is simulated for 20 seconds, while the FitzHugh-Nagumo and Coupled Rössler models are simulated for 50 seconds, respectively. All simulation trajectories are downsampled at equal intervals to obtain 500 observations. The dataset is then divided into training, validation, and testing sets in a ratio of 6:2:2.

B SOFTWARE AND HARDWARE ENVIRONMENT

We implement DiskNet in PyTorch and employ the open-source available implementations with default parameters for baselines. All experiments were conducted on the NVIDIA GeForce RTX 4090 GPU. For all datasets, we set the batch size to 8 and trained for 50 epochs with a learning rate of 0.001.

C PSEUDO-CODE OF THE PRE-TRAINING PHASE

We perform supervised pre-training on the Assignment Matrix Calculation module as described in Algorithm 1. The supervisory signal is given by the assignment matrix P_0 using the RG method from statistical physics.

Algorithm 1 Physics-informed Pre-training

Require: hyperbolic embedding C^H and C_s^H;
Ensure: pre-trained MLPs with weights θ_1 and θ_2
1: calculate assignment matrix P_0 by physical knowledge;
2: convert (C^H, C_s^H) to (C^E, C_s^E) using equation (3);
3: **for** i=1; i ≤ n; i++ **do**
4:     node representation $\tilde{C} = MLP_{\theta_1}(C^E)$;
5:     super-node representation $\tilde{C}_s = MLP_{\theta_2}(C_s^E)$;
6:     assignment $P = \text{softmax}(\tilde{C}_s \tilde{C}^T)$;
7:     loss $L_p = \frac{1}{\gamma N} \sum |P - P_0|$;
8:     compute gradient directions ∇L_p;
9:     update θ_1 and θ_2;
10: **end for**

D MORE EXPERIMENT RESULTS

D.1 Identified Skeletons in Main Experiments

In this section, we visualize the identified skeletons for all real-world networks in Table 1 for different dynamics with $\gamma = 50\%$: (a) Drosophila network in Figure 8, (b) Airport network in Figure 9,(c) Social network in Figure 10,(d) Web network in Figure 11, and (e) PowerGrid network in Figure 12. As discussed in Sec. 4.3.2, there are significant differences in the skeletons of networks with the same topology but different dynamics, and the assignment patterns exhibit a high correlation with both angular and radial coordinates.
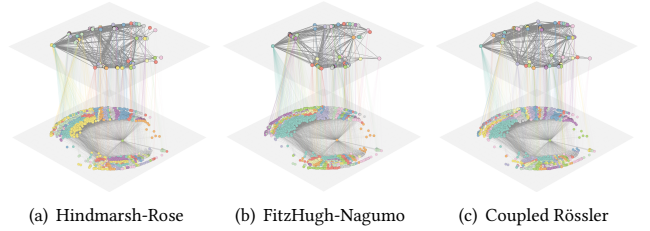

(a) Hindmarsh-Rose   (b) FitzHugh-Nagumo   (c) Coupled Rössler

Figure 8: Identified skeletons of the Drosophila network.

(a) Hindmarsh-Rose   (b) FitzHugh-Nagumo   (c) Coupled Rössler
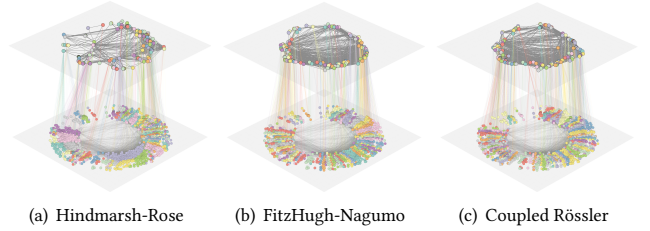
Figure 9: Identified skeletons of the Airport network.

(a) Hindmarsh-Rose   (b) FitzHugh-Nagumo   (c) Coupled Rössler
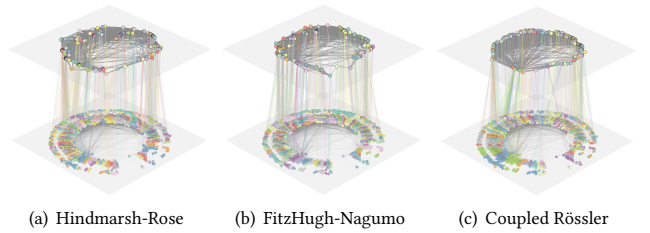
Figure 10: Identified skeletons of the Social network.

(a) Hindmarsh-Rose   (b) FitzHugh-Nagumo   (c) Coupled Rössler
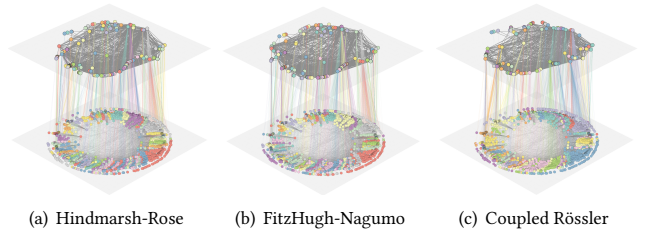
Figure 11: Identified skeletons of the Web network.

(a) Hindmarsh-Rose   (b) FitzHugh-Nagumo   (c) Coupled Rössler
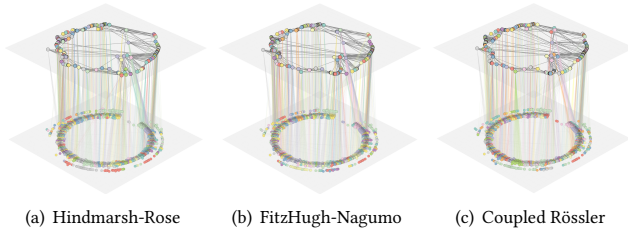
Figure 12: Identified skeletons of the PowerGrid network.