Let Brain Rhythm Shape Machine Intelligence for Connecting Dots on Graphs

Jiaqi Ding[1]* **Tingting Dan**[2]* **Zhixuan Zhou**[1] **Guorong Wu**[1,2]†
[1]Department of Computer Science [2]Department of Psychiatry
The University of North Carolina at Chapel Hill
{jiaqid, zzhixuan}@cs.unc.edu {Tingting_Dan, grwu}@med.unc.edu

"The whole universe is a complex system, including the human brain." – Immanuel Kant

Abstract

In both neuroscience and artificial intelligence (AI), it is well-established that neural "coupling" gives rise to dynamically distributed systems. These systems exhibit self-organized spatiotemporal patterns of synchronized neural oscillations, enabling the representation of abstract concepts. By capitalizing on the unprecedented amount of human neuroimaging data, we propose that advancing the theoretical understanding of rhythmic coordination in neural circuits can offer powerful design principles for the next generation of machine learning models with improved efficiency and robustness. To this end, we introduce a physics-informed deep learning framework for Brain Rhythm Identification by Kuramoto and Control (coined *BRICK*) to characterize the synchronization of neural oscillations that shapes the dynamics of evolving cognitive states. Recognizing that brain networks are structurally connected yet behaviorally dynamic, we further conceptualize rhythmic neural activity as an artificial dynamical system of coupled oscillators, offering a shared mechanistic bridge to brain-inspired machine intelligence. By treating each node as an oscillator interacting with its neighbors, this approach moves beyond the conventional paradigm of graph heat diffusion and establishes a new regime of representation compression through oscillatory synchronization. Empirical evaluations demonstrate that this synchronization-driven mechanism not only mitigates over-smoothing in deep GNNs but also enhances the model's capacity for reasoning and solving complex graph-based problems.

1 Introduction

The evolution of artificial intelligence (AI) has long been intertwined with efforts to decipher the principles of human cognition. Rather than existing as separate disciplines, AI and neuroscience share conceptual foundations in how information is represented and transformed. In the human brain, massive ensembles of neurons form densely interconnected circuits that coordinate activity across multiple spatial and temporal scales [4]. Many artificial graph systems, ranging from transportation networks to online communities, show comparable organizational motifs, including efficient long-range connectivity and modular substructures [57].

Both domains can be viewed through the lens of collective dynamics. Neural assemblies give rise to oscillatory rhythms through intricate coupling between regions [20], while graph neural

*Equal contribution.
†Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

networks (GNNs) update node states by iteratively exchanging information, leading to structured representations that emerge across layers [58].

Machine learning shows remarkable proficiency in extracting statistical patterns from vast datasets, yet it remains fundamentally limited in its capacity for flexible reasoning and context-aware integration. Biological neural systems, in contrast, constantly fuse inputs from different sensory streams in real time [24]. Everyday experiences, for example, hearing approaching footsteps before seeing someone, highlight how sensory predictions interact across modalities to construct unified percepts.

Neural oscillations serve as a core mechanism that enables coordinated information exchange across different sensory modalities [53, 49]. At the microscopic scale, accumulating evidence indicates that neurons communicate through lateral connections [31], and their rhythmic patterns of excitability, commonly referred to as brain rhythms, emerge from the intrinsic dynamics of local circuits and the biophysical properties of ionic channels [8]. Neighboring neurons often synchronize their activity, forming competitive clusters that interpret sensory inputs [25, 45]. This process, known as "competitive learning" [2], compresses information during layer-wise propagation and enhance abstraction. Such synchronization also drives functional specialization among higher-order cortical populations, aligning their activity (analogous to fireflies synchronizing their flashes) to generate compact and meaningful neural representations [7].

While modern GNNs such as graph convolution network (GCN) [34] and Graph Transformer [63] excel at structural representation learning, they lack the adaptive dynamical properties that characterize biological neural systems. On the flip side, cognition in humans and animals is governed by large-scale, dynamic oscillations that support perception, memory and decision-making. This parallel between biological and artificial systems motivates us to rethink graph learning through the lens of coupled neural oscillators, which presents new opportunities to reshape GNN architectures.

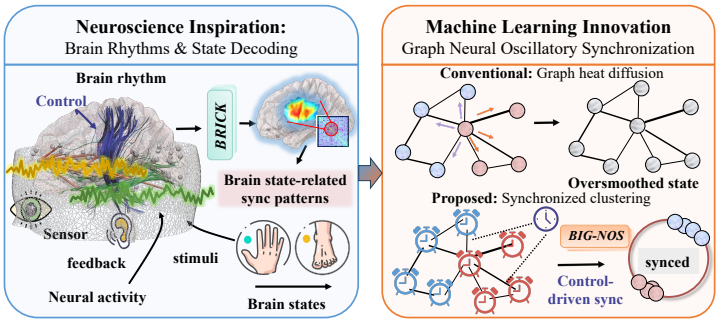

Figure 1: The link between the human brain and artificial intelligence inspires the development of novel learning mechanism of graph-structured data based on neural oscillatory synchronization.

Following this spirit, we aim to establish a novel *learning mechanism for graph-structured data* grounded in neural oscillatory synchronization, as illustrated in Fig. 1. ***First***, we introduce *BRICK*, a physics-informed deep learning framework for brain rhythm identification. This model unifies the synchronization dynamics of the Kuramoto model [36] with the concept of attending memory [32], allowing us to capture rhythmic neural oscillations that underlie cognitive state transitions. By bridging control theory with oscillatory dynamics, *BRICK* provides a principled foundation for extracting interpretable task-related synchronization patterns from complex neural signals. ***Second***, we generalize this oscillatory synchronization principle to graph learning and propose *BIG-NOS*, which is a biologically inspired graph neural oscillatory synchronization framework. Here, each graph node functions as a coupled oscillator whose state evolves dynamically through network interactions. In contrast to conventional graph message passing and convolution paradigms constrained by heat diffusion [11], which often suffer from over-smoothing and information dilution [37, 17], *BIG-NOS* preserves oscillatory coherence across graph topology. By steering feature propagation via controlled synchronization, the model naturally captures emergent interference patterns in the spatial domain, unveiling meaningful modular structures (e.g., red–blue clustering in Fig. 1). This biologically grounded formulation draws inspiration from how rhythmic synchronization supports information integration in the human brain, enriching the theoretical connection between brain dynamics and graph learning and offering a powerful foundation for next-generation machine intelligence, capable of interpretable reasoning, robust pattern integration, and scalable performance on real-world graph datasets. In summary, we propose a neural oscillatory synchronization-based graph learning mechanism, uniting a physics-informed model (*BRICK*) and a biologically grounded GNN (*BIG-NOS*) with decent real-world performance.

2 Related Works

Graph neural networks for brain network modeling. GNNs have become foundational tools for analyzing structured data, spanning areas such as chemistry, recommendation systems, biological networks, and social graphs. Initial developments focused on spectral formulations like GCN [34] and GAT [56], which were later extended by more expressive variants including GIN [59] and GraphSAGE [27]. Beyond standard tasks of node and graph classification, GNNs have been adapted for spatio-temporal modeling [60], link prediction, and graph generation [38], with scalability improved through methods such as clustering and sparse attention [15, 61, 35].

Applications to brain networks have followed a similar trajectory, using both structural and functional connectomes to represent the complex topology of neural systems. Early work such as BrainNetCNN [33] demonstrated the feasibility of disease classification from brain graphs, while more recent studies leverage hierarchical pooling [62] and attention mechanisms [29] to uncover multi-scale functional organization and enhance interpretability. Temporal extensions [60] have further enabled dynamic modeling of fMRI and EEG signals, offering insight into evolving neural states over time.

System identification approaches. A key objective in computational neuroscience has been to uncover mechanistic principles governing how the brain regulates perception, action, and cognition [6]. Motivated by parallels between cognitive control and classical control theory, prior work has represented neural dynamics as linear or bilinear control systems [26, 40], enabling controllability analyses through energy-based formulations [41]. More recently, research has shifted toward integrating recurrent neural architectures with physics-inspired continuous-time dynamics [14, 28, 16]. The Kuramoto model, in particular, has been a central tool for linking structural connectivity to emergent functional activity [9, 10], showing that spontaneous neural fluctuations can be explained as coordinated interactions between local oscillators and large-scale network architecture. Our proposed *BRICK* builds on this line of work by parameterizing these governing dynamics through a deep learning framework.

Kuramoto dynamics in machine learning. A closely related effort is the recent development of artificial Kuramoto oscillatory neurons for unsupervised vision representation [43], which leverage synchronization to achieve compact and structured feature embeddings. In contrast, our approach emphasizes the bidirectional exchange between neuroscience and machine learning through dynamical systems principles [18], and places additional focus on probing the intrinsic potential of the model. Specifically, we aim to (1) use machine learning models to reveal the mechanisms by which fluctuating brain activity maps to cognitive states, and (2) translate biologically grounded synchronization mechanisms into graph learning to enable large-scale network reasoning without explicit supervision.

3 Methods

In this work, we model both human brain networks and graph-structured data as dynamical systems whose collective behavior is governed by the Kuramoto model [18]. Consider a network of N interacting oscillators with pairwise coupling strengths K_{ij} $(i, j = 1, \ldots, N)$. Each oscillator oscillates at its intrinsic frequency ω_i, and the temporal evolution of its phase θ_i can be expressed as

$$\frac{d\theta_i}{dt} = \omega_i + \sum_{j=1}^{N} K_{ij} \sin\big(\theta_j - \theta_i\big), \tag{1}$$

where $\theta_i \in \mathbb{R}$ denotes the instantaneous phase of the i-th oscillator. When coupling is absent, the oscillators evolve independently following their own natural frequencies. As interactions accumulate over time, the sinusoidal coupling term in Eq. 1 drives gradual phase alignment across units, giving rise to collective synchronization. This interaction enables groups of oscillators to entrain to common rhythms, ultimately forming stable phase-locked clusters (as shown in Fig. 1 bottom-right).

3.1 Deep Model for Brain Rhythm Identification

A growing body of research has highlighted the central role of neural oscillations in coordinating activity across distributed brain regions [8, 55]. To mechanistically characterize these large-scale synchronization phenomena, the Kuramoto model has become a widely used framework in both theoretical neuroscience and neuroimaging research [9].

Problem setup. We formalize the brain network as a weighted graph $\mathcal{G} = (V, W)$, where V represents N brain regions (nodes) and $W = [w_{ij}]_{i,j=1}^{N}$ denotes the weighted adjacency matrix. Each weight w_{ij} encodes the coupling intensity between node v_i and v_j, derived from neuroimaging measurements[3]. Let the observed BOLD signal be $X = x_i \mid x_i(t) \in \mathbb{R}, t = 1, \ldots, T$, where $x_i(t)$ is the signal measured from brain region v_i at time t. We then construct a physics-informed deep learning model to map neural activity patterns to cognitive outcomes.

Vectorized neural oscillators. To move beyond scalar-valued signals $x_i(t)$ and better characterize complex neural dynamics, we adopt the geometric scattering transform (GST) [23]. GST leverages harmonic wavelets derived from the graph Laplacian to construct representations that encode both multi-scale structure and frequency content. We begin by defining the lazy random walk matrix [42]: $G = \frac{1}{2}(I_N + WD^{-1})$, where I_N denotes the $N \times N$ identity matrix and D is the diagonal degree matrix associated with W. From this operator, a sequence of graph wavelets $\{\Psi_h\}_{h=0}^{H-1}$ is generated recursively, enabling a compact and structured embedding of neural oscillatory activity across multiple scales. $\Psi_0 := I_N - G, \quad \Psi_h := G^{2^{h-1}} - G^{2^h}, \quad 1 \leq h < H$. Let $x(t) \in \mathbb{R}^N$ be the BOLD snapshot at time t. For each scale h, the GST output $\hat{x}^h(t) \in \mathbb{R}^N$ is computed by: (1) applying the wavelet transform $(\Psi_h, x(t))$, (2) taking element-wise absolute value, and (3) applying the low-pass filter $\Phi = G^{2^H}$, yielding $\hat{x}^h(t) = \Phi|(\Psi_h, x(t))|$.

Vectorized Kuramoto model for multi-frequency neural synchronization. Following GST decomposition, each brain region v_i yields a set of frequency-resolved BOLD signals $\hat{x}_i(t) = [\hat{x}_i^h(t)]_{h=1}^{H}$ capturing temporal fluctuations across multiple scales. The evolution of these oscillator phases can be compactly described by a vector-valued differential equation:

$$\frac{d\hat{x}_i}{dt} = \omega_i + [\tau \cdot \phi_{\hat{x}_i}(\sum\nolimits_{j=1}^{N} w_{ij}\hat{x}_j)] \tag{2}$$

where $\hat{x}_i \in \mathbb{R}^H$ denotes the oscillator state of region v_i on a unit hypersphere, ω_i its intrinsic frequency, and τ a global coupling gain parameter. The mapping ϕ constrains the aggregated phase input to the tangent space of $\hat{x}_i$, enforcing smooth manifold-constrained dynamics.

Rather than simply fitting temporal fluctuations, the aim is to represent neural synchronization in a way that reflects how the brain achieves coordinated oscillatory activity with minimal energetic cost. Accordingly, we seek an oscillator-based representation and a control strategy that are both mathematically stable and biologically meaningful.

Kuramoto model with attending memory. The standard Kuramoto model (Eq. 2) relies on global dynamics, which are not well suited for capturing short-lived functional fluctuations tied to specific cognitive operations. In contrast, the human brain can flexibly emphasize task-relevant inputs while down-weighting irrelevant signals, effectively deciding "what to remember" and "what to ignore." This selective filtering resonates with the concept of attending memory [32] and motivates us to augment the Kuramoto model with a biologically inspired, memory-guided control mechanism. We incorporate a global optimal control term:

$$\frac{d\hat{x}_i}{dt} = \omega_i + [\tau \cdot \phi_{\hat{x}_i}(y_i + \sum\nolimits_{j=1}^{N} w_{ij}\hat{x}_j)], \tag{3}$$

where y_i is a feedback control signal derived from attending memory, reflecting observed neural activity. Each y_i represents behavior-specific, population-level memory traces that dynamically modulate the influence of oscillator i in a given cognitive context. This control formulation allows the system to continuously adapt synchronization dynamics as neural states evolve.

Intuition behind the Kuramoto model with attending memory. Assuming uniform contribution from each brain region v_i and a symmetric coupling matrix W, the extended Kuramoto dynamics in Eq. 3 can be viewed as a gradient flow that evolves along the steepest descent direction of the following energy functional E [3, 43]:

$$E = -\sum_{i,j} \hat{x}_i^\top w_{ij} \hat{x}_j - \sum_i y_i^\top \hat{x}_i \tag{4}$$

The term E acts as a Lyapunov function that guarantees the system's stability, as $\frac{dE(\hat{x}(t))}{dt} \leq 0$. Its first component captures pairwise coupling, promoting synchronization among oscillators $\hat{x}_i$ and $\hat{x}_j$

[3]For example, w_{ij} can represent the normalized fiber count in structural connectivity (SC) or the strength of temporal co-fluctuation in functional connectivity (FC).

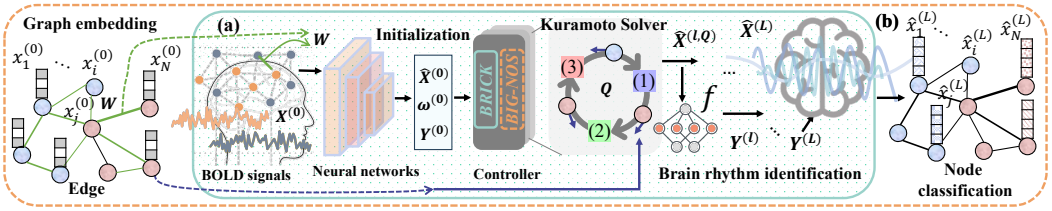

Figure 2: Overview of the proposed framework. (a) *BRICK* (solid cyan box) models neural oscillatory synchronization for brain rhythm identification. (b) *BIG-NOS* (dashed orange box) extends the same Kuramoto-based dynamics to graph learning.

proportional to the interaction strength w_{ij}. The second component represents task-driven feedback control, aligning each oscillator's dynamics with its control signal y_i, which encodes population-level attention and behavior-specific modulation. Together, these two objectives describe a biologically motivated trade-off between network coherence and cognitive specialization.

From a control-theoretic standpoint, the feedback variables $Y = \{y_i | i = 1, ..., N\}$ can be interpreted as spatial control fields, providing a physically grounded attention mechanism distinct from graph attention [56]. These feedback patterns quantify the controllability of each brain region [39], bridging the link between oscillatory synchronization and adaptive regulation of cognitive states. Importantly, the energy formulation also offers a gradient-flow interpretation, ensuring that the system monotonically decreases E along its trajectory, thereby maintaining smooth and stable evolution on the hyperspherical manifold.

This unified view reveals why the proposed model alleviates the over-smoothing problem observed in deep GNNs: oscillatory synchronization fosters global coherence while preserving discriminative, high-frequency information. Consequently, this mechanism provides both a neurobiologically meaningful and computationally efficient pathway toward large-scale graph reasoning, forming the theoretical basis of the *BIG-NOS* model introduced in Sec. 3.2.

Physics-informed deep model for brain rhythm identification. Building upon the above formulation, we introduce *BRICK*, a physics-informed framework designed to predict cognitive states through neural synchronization dynamics. The overall architecture of the model is illustrated in the solid cyan box in Fig. 2a. Specifically, the intrinsic frequency of each oscillator is determined by a set of learnable vectors $\Gamma_\sigma(\hat{x}_i)$, whose magnitudes control the rotation speed, while the control signal y_i is generated through a neural mapping $\Gamma_\mu(\hat{x}_i)$, with σ and μ denoting network parameters. The synchronization process evolves in three key steps: **(1) Coupling influence aggregation.** For each oscillator $\hat{x}_i$, we compute the aggregate input from its neighbors as $z_i = y_i + \sum_{j=1}^{N}(s_{ij} \cdot w_{ij})\,\hat{x}_j$, where $S = [s_{ij}]_{i,j=1}^{N}$ is a symmetric, trainable reweighting matrix that adaptively modulates the coupling strengths W. This step captures how local interactions shape the global phase landscape. **(2) Tangent-space projection.** Each z_i is then projected onto the tangent space of the unit hypersphere at $\hat{x}_i$ through $\phi_{\hat{x}_i}(z_i) = z_i - \langle z_i, \hat{x}_i \rangle \hat{x}_i$. This ensures that the updated direction respects the geometric constraint of the manifold and maintains stability during the synchronization process [12]. **(3) Phase-state update.** Finally, $\hat{x}_i$ is updated via forward Euler integration of Eq. 3, while the controller y_i is transformed using a mapping function f_φ to remain on the hypersphere and capture phase-invariant patterns. These three steps together implement a geometry-aware synchronization mechanism in which oscillator states evolve under both coupling dynamics and task-driven control. The procedure is summarized in Algorithm 1.

The downstream objective combines the task-driven loss $\mathcal{L}$, defined as cross-entropy between predicted and ground-truth cognitive task labels, with the synchronization energy E. For unseen subjects, the trained *BRICK* synchronizes a large oscillator ensemble $\hat{X}$ using learned parameters σ and μ, producing both the predicted cognitive state and its associated control pattern Y.

3.2 *BIG-NOS*: Graph Learning via Neural Oscillation

New learning mechanism of GNN. Conceptually, *BIG-NOS* extends *BRICK* from continuous brain dynamics to arbitrary graph data (as shown in the dashed orange box of Fig. 2b). By treating graph nodes as analogs of brain regions and the adjacency matrix as a proxy for inter-regional coupling

Algorithm 1: Iterative Solver for *BRICK* Dynamics (Eq. 3)

Input: BOLD signal $X^{(0)}$, coupling matrix W

Output: Final oscillator states $\hat{X}^{(L)}$ and controller $Y^{(L)}$

Initialization: Obtain initial oscillator states $\hat{X}^{(0)}$ by applying GST to $X^{(0)}$; set $Y^{(0)} = \Gamma_\mu(X^{(0)})$;

Parameterize natural frequencies $\Omega \leftarrow \Gamma_\sigma(\hat{X}^{(0)})$;

for $l = 1$ **to** L **do**

  for $q = 1$ **to** Q **do**

    `// Step 1: Aggregate coupling influence`

    $Z \leftarrow Y^{(l)} + (S \odot W)\hat{X}^{(q)}$;

    `// Step 2: Project to tangent space of the hypersphere`

    $\phi_{\hat{X}^{(q)}}(Z) \leftarrow Z - \langle Z, \hat{X}^{(q)} \rangle \hat{X}^{(q)}$;

    `// Step 3: Integrate oscillator dynamics`

    $\Delta\hat{X}^{(q)} \leftarrow \Omega + \tau \cdot \phi_{\hat{X}^{(q)}}(Z)$;

    $\hat{X}^{(q+1)} \leftarrow \hat{X}^{(q)} + \beta \cdot \Delta\hat{X}^{(q)}$;

    `// Step 4: Renormalize to stay on manifold`

    $\hat{X}^{(q+1)} \leftarrow \hat{X}^{(q+1)} / \|\hat{X}^{(q+1)}\|$;

  end

  `// Step 5: Update control signal on hypersphere`

  $Y^{(l+1)} \leftarrow f_\varphi(\hat{X}^{(q+1)})$;

end

strength, we seamlessly adapt the physics-informed architecture of *BRICK* to general graphs. Both models share the same governing equation rooted in Kuramoto synchronization but differ in their data domains and learning objectives.

Let $X = [x_i]_{i=1}^N$ denote the initial graph embeddings. In conventional GNNs, feature propagation follows a graph heat diffusion process, formalized as $\frac{\partial X}{\partial t} = \nabla \cdot (\nabla X)$, where ∇ and $\nabla\cdot$ denote the graph gradient and divergence operators [11]. Prior studies have shown that excessive message passing under this diffusion mechanism leads to over-smoothing, where node representations become indistinguishable [37]. Inspired by *BRICK*, we mitigate over-smoothing by evolving graph features within a latent oscillatory phase space rather than diffusing them over the graph domain. The Kuramoto model's intrinsic oscillatory dynamics, driven by coupling interactions and intrinsic frequencies, naturally prevent convergence to static equilibrium (as in diffusion-based GNNs [51]) and instead produce partial synchronization [1].

Unlike conventional attention mechanisms that re-weight neighbor contributions during aggregation [56], our model introduces *control patterns* Y as task-dependent feedback signals that guide synchronization dynamics (Eq. 3). These patterns provide a new perspective on graph controllability through the lens of complex systems theory [39].

Taken together, conventional GNNs are typically governed by a diffusion equation, in which node features evolve according to a heat-diffusion process. This formulation leads to passive information averaging across neighboring nodes, an approach that effectively captures local smoothness but often suffers from over-smoothing and limited expressiveness in deeper architectures. In contrast, our brain-rhythm-inspired *BIG-NOS* is formulated as a neural oscillation system, where each node behaves as a coupled oscillator whose state evolves according to both its intrinsic frequency and the phase interactions with connected nodes. Rather than diffusive averaging, information propagation in *BIG-NOS* emerges from oscillatory synchronization, allowing it to represent complex, nonlinear dependencies across the network. Furthermore, unlike conventional attention mechanisms that rely on fixed weighting matrices, *BIG-NOS* incorporates a task-adaptive feedback control mechanism that dynamically modulates coupling strength, enabling flexible coordination and improved generalization across diverse graph structures.

Network architecture. We discretize the *BRICK* formulation into a GNN framework, denoted as *BIG-NOS*, to enable general graph learning. In this architecture, each node is represented as an

oscillator, and its interactions are governed by the underlying graph topology. The model takes initial graph embeddings $X^{(0)}$ as input, and an encoder (e.g., a GCN layer) is used to initialize the control term. As illustrated in Fig. 2b, feature evolution in *BIG-NOS* follows Kuramoto-based dynamics, which are explicitly described in Algorithm 1.

4 Experiments

In this section, we present extensive experiments to validate the effectiveness and interpretability of our proposed models across a variety of brain-related and general graph learning tasks.

We conduct a comprehensive evaluation of our proposed models across multiple tasks. We evaluate *BRICK* for brain rhythm identification and explore its effectiveness in unsupervised brain parcellation, demonstrating its potential for uncovering latent neural patterns in neural dynamics. We further evaluate *BIG-NOS* on standard benchmarks including node and graph classification. To analyze its robustness to "over-smoothing", we examine performance across increasing network depths. We compared *BIG-NOS* with a diverse set of graph-based baselines: GCN [34], GAT [56], GIN [59], GCNII [13], GraphSAGE [27], SAN [35], GRAND [11], GTN [63], GraphCON [48] and KuramotoGNN (KGNN) [46]. To validate scalability, we also evaluate *BIG-NOS* on the large-scale ogbn-arxiv dataset. For this task, we compare not only against the above baselines but also with top-performing models from the official OGB leaderboard.

4.1 Data Description

To evaluate the effectiveness of *BRICK* and *BIG-NOS*, we use the publicly available neuroimaging and graph datasets, respectively.

• Human brain datasets

1. HCP-Aging (HCP-A) [5]. This dataset includes 717 subjects with 4,846 fMRI scans (300 time points each), covering four tasks (VISMOTOR, CARIT, FACENAME, Resting State).

2. HCP-Young Adults (HCP-YA) [54]. This includes seven cognitive tasks: Motor, Relational, Social, Working Memory, Language, Emotion, and Gambling (175 time points each). The Working Memory task (HCP-WM) involves alternating 2-back and 0-back conditions with body, place, face, and tool stimuli with a total of 405 time points. Preprocessing follows [19].

In both HCP-A and HCP-YA datasets, the brain is parcellated into 116 regions via the AAL atlas [52]. Structural connectivity (SC) matrices with the size of 116×116 encode fiber counts between regions (normalized per subject), while functional connectivity (FC) is derived from Pearson correlations of BOLD signals across brain regions. To test scalability, we use the Brainnetome atlas [22] to increase granularity to 246 regions for HCP-WM. We perform 5-fold cross-validation for both the 4-class (HCP-A), 7-class (HCP-YA) and 8-class (HCP-WM) classification tasks.

• Graph-based datasets

1. Node classification. We apply *BIG-NOS* to homophilic and heterophilic graphs sorted by homophily ratio h [64]: Texas (h=0.11), Wisconsin (0.21), Actor (0.22), Squirrel (0.22), Chameleon (0.23), Cornell (0.3), Citeseer (0.74), Pubmed (0.80), Cora (0.81). We also evaluate on the large-scale **ogbn-arxiv** dataset from the Open Graph Benchmark (OGB) [30]. Details are in Table 4 (Appendix).

2. Graph classification. We evaluate on ENZYMES and PROTEINS from TUDataset [44], detailed in Table. 5. For homophilic graph data (Cora, Citeseer, Pubmed) in node classification tasks, we adopt the semi-supervised 20-per-class training split from [34] and average over 5 random seeds. For heterophilic graph data and TUDataset, we follow [47] and [21], using 10-fold cross-validation and reporting test-set averages. All datasets are evaluated using Accuracy (Acc), weighted-Precision (Pre) and F1-Score (F1).

4.2 Performance on Human Brain Data

4.2.1 Brain rhythm identification

Results. We evaluate *BRICK* on the task of decoding dynamic brain states through neural synchronization. Fig. 3a reports the performance across three human brain datasets and seven baseline

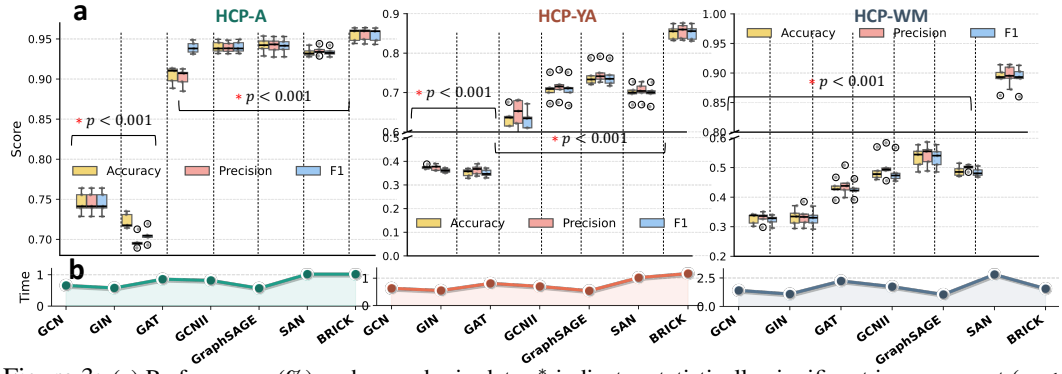

Figure 3: (a) Performance (%) on human brain data. * indicates statistically significant improvement ($p < 0.001$). (b) Inference time (ms/subject) on three human brain datasets.

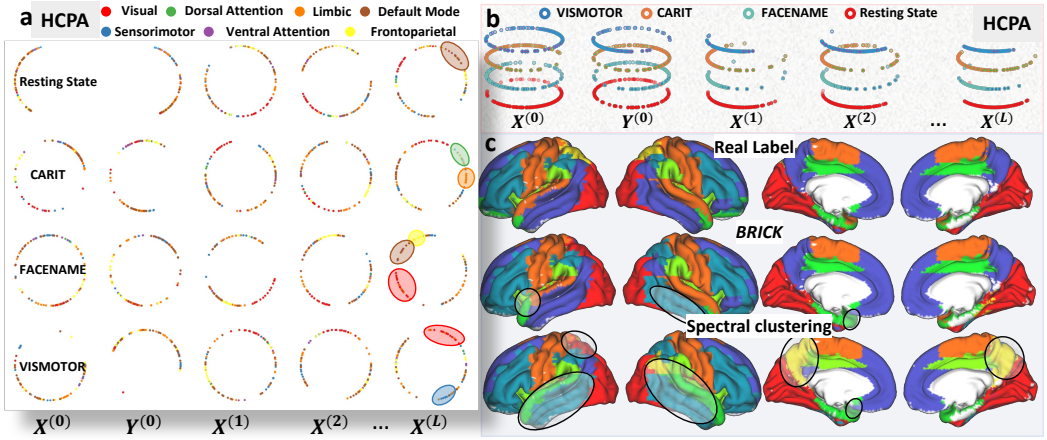

Figure 4: (a) Distribution of regional features across different brain states (tasks) in the HCP-A dataset. Each brain region's feature trajectory over time is projected onto the unit phase space (magnitude = 1), revealing consistent synchronization of the same regions across tasks. (b) Phase-space feature aggregation on HCP-A dataset, where features from different categories at each time point are mapped onto the unit phase space to illustrate their clustering patterns. (c) Unsupervised clustering comparison between *BRICK* and classical spectral clustering, where *BRICK* yields more coherent and neurobiologically meaningful functional communities.

models. *BRICK* consistently achieves the best performance ($p < 0.001$), surpassing all existing hand-designed GNN models.

Discussion. These results demonstrate that our physics-based oscillation model effectively synchronizes brain regions, producing patterns closely aligned with specific cognitive states. To further explore this relationship, we visualize the phase-space representations of brain regions in Fig. 4a. *BRICK* captures task-specific synchronization across functionally defined brain network in HCP-A. For instance, Visual network regions (red) cluster clearly during the VISMOTOR task in the deeper representation $X^{(L)}$, indicating strong within-network coordination. Similarly, tighter phase clustering emerges in the Sensorimotor and Dorsal Attention networks under tasks such as CART and FACENAME. These results suggest that our model respects functional boundaries in a biologically meaningful manner.

We also investigate task-specific synchronization across all subjects. As shown in Fig. 4b, individuals performing the same task (e.g., "VISMOTOR" or "FACENAME" in HCP-A dataset) form well-separated synchronization in deeper layers (e.g., $X^{(L)}$). This indicates *BRICK* captures latent dynamics consistent across subjects and datasets, supporting the notion that neural phase alignment underlines cognitive state representation.

4.2.2 Potential in unsupervised brain parcellation

To evaluate our model's capability in unsupervised settings, we examine functional parcellation on the HCP-A dataset, which partitions 116 brain regions (AAL atlas) into eight functional subnetworks

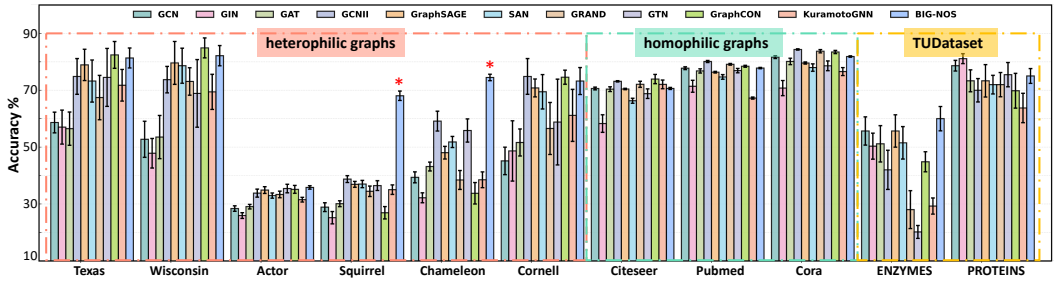

Figure 5: Accuracy (%) on node and graph classification tasks. * indicates significant improvement ($p < 0.001$).

(default mode, frontoparietal, limbic, ventral/dorsal attention, sensorimotor, visual, and cerebellar networks). We train an *unsupervised* variant of *BRICK* that identifies functional communities by minimizing pairwise distances between synchronized oscillators x_i and x_j using a Rayleigh quotient objective [50]. As shown in Fig. 4c, compared to classical spectral clustering, *BRICK* more effectively synchronizes related brain regions into meaningful clusters, demonstrating strong potential for personalized brain parcellation.

4.2.3 Ablation study and model efficiency

We conduct an ablation study on the GST to assess the impact of different frequency scales. Specifically, we vary the wavelet order (from 0 to 2) and level (1 or 2) on the HCP-YA dataset (fixed-group split), which define a multi-branch wavelet tree: the level determines the tree's depth, while the order sets the branching factor. As shown in Table 1, increasing either the level or order generally improves performance by incorporating additional frequency components. However, this comes at the cost of increased computational complexity. For example, using `level=2` and `order=[0,1]` expands the feature dimension from $[N]$ to $[N, 7]$, due to the increased number of wavelet filters.

Table 1: Ablation results of GST with different levels and orders on HCP-YA.

Level	Order	Acc (%)	Pre (%)	F1 (%)
1	**[0]**	-0.41	-0.45	-0.42
	[0, 1]	-0.14	-0.05	-0.16
	[0, 1, 2]	85.20	85.53	85.13
2	**[0]**	-0.23	-0.24	-0.24
	[0, 1]	+0.21	+0.25	+0.22

We provide a detailed comparison of inference time per subject across three human brain datasets in Fig. 3b. Overall, all baseline GNNs demonstrate low computational cost, with inference typically below 2 ms per subject. We further PDE-based GRAND and KuramotoGNN are notably slower. In contrast, *BRICK* achieves comparable efficiency to lightweight baselines such as GCN and GraphSAGE. Even on the larger HCP-WM dataset (246 regions), *BRICK* maintains a modest runtime of 1.56 ms per subject (as shown in Fig. 3b-the rightmost data point), confirming its scalability and computational practicality for large-scale brain analysis.

4.3 Performance on Graph Data

4.3.1 Node classification

Result. Fig. 5 (left and middle) presents the results of eleven models across nine node-classification benchmarks. Overall, *BIG-NOS* achieves the best or second-best accuracy on the majority of datasets, showing strong generalization across both homophilic and heterophilic graphs. For homophilic graphs (*Citeseer*, *Pubmed*, *Cora*), *BIG-NOS* maintains competitive performance on par with advanced architectures like GCNII and GRAND. For heterophilic graphs, it achieves the highest performance on *Actor*, *Squirrel*, and *Chameleon*, where conventional message-passing GNNs typically fail due to inconsistent neighborhood features. In contrast, *BIG-NOS* continues to improve through phase-based coupling, which allows nodes to synchronize dynamically rather than relying solely on feature similarity. This oscillatory mechanism enables coherent message propagation even when node attributes are diverse, making *BIG-NOS* especially suitable for heterophilic settings.

Scalability. To assess the scalability of *BIG-NOS*, we evaluated its performance on the large-scale ogbn-arxiv dataset. As reported in Table 2, our model achieved an accuracy of **0.86**, outperforming

Table 2: Performance on **ogbn-arxiv** test set. **Bold** and underline indicate the best and second-best results.

Dataset	Metric	GCN	GIN	GAT	GCNII	GraphSAGE	SAN	BiGTex[a]	SimTeG+TAPE+RevGAT[a]	***BIG-NOS***
ogbn-arxiv	Acc	0.70	0.71	0.70	0.71	0.71	0.69	**0.89**	0.78	0.86
	Pre	0.70	0.70	0.69	0.71	0.70	0.68	–	–	**0.86**
	F1	0.69	0.69	0.68	0.70	0.69	0.68	–	–	**0.85**

[a] *BiGTex* and *SimTeG+TAPE+RevGAT* are the top two models on the OGB Leaderboard at the time of writing. Pre and F1 are not reported.

all standard baselines and ranking second on the leaderboard, just behind BiGTex (0.89). These results underscore the strong generalization ability of our oscillatory synchronization mechanism on large graphs.

Discussion. These results show that our message aggregation strategy effectively promotes synchronization among nodes of the same class while preserving the structural dependencies defined by the coupling strength. In contrast to conventional GNNs, which tend to drive neighboring nodes toward indistinguishable embeddings, our model maintains representational diversity and prevents excessive feature homogenization. As a result, it inherently resists over-smoothing as the network depth increases. This property is empirically verified in Fig. 6, where we observe consistently high classification performance even with up to 128 layers, demonstrating the robustness of our synchronization framework against the depth-induced degradation common in traditional GNNs.

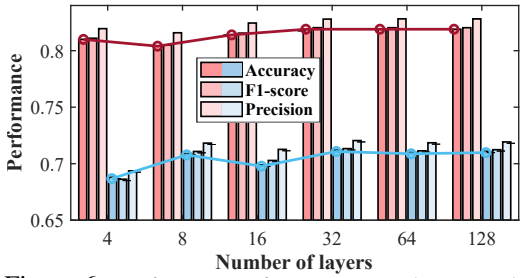

Figure 6: Performance of *BIG-NOS* as the network depth increases on *Cora* (red) and *Citeseer* (blue).

4.3.2 Graph classification

Result. Fig. 5 (right) reports graph classification accuracy on the ENZYMES and PROTEINS datasets from TUDataset [44]. *BIG-NOS* achieves the best accuracy on ENZYMES ($60.00_{\pm 4.28}$) and performs competitively on PROTEINS ($75.02_{\pm 2.61}$), despite using a minimal batch size (batch size=1). These results indicate that our model maintains strong generalization from node-level to graph-level tasks without architectural modification.

Discussion. Graph classification is conceptually analogous to brain rhythm identification, where the goal is to predict a global state emerging from local interactions. The success of *BIG-NOS* in this setting further validates the ability of our synchronization mechanism to generate global patterns that effectively distinguish different graphs, which reinforces its applicability beyond node-level tasks.

Table 3 summarizes the average accuracy across homophilic, heterophilic, and all datasets. The balanced performance is particularly encouraging: whereas many existing models tend to specialize in a single regime (for example, GCN performs well primarily on homophilic graphs), our model demonstrates consistent generalization across both homophilic and heterophilic graphs. This robustness suggests that the underlying neural oscillation mechanism is inherently generalizable and largely independent of specific graph structures.

Table 3: The average accuracy across homophilic, heterophilic, and all datasets.

	GCN	GIN	GAT	GCNII	GraphSAGE	SAN	GRAND	GTN	GraphCON	KGNN	***BIG-NOS***
Hete. Avg	42.19	39.46	43.99	62.01	58.19	57.18	50.55	54.99	56.28	51.23	**69.18**
Homo. Avg	76.68	66.81	75.77	79.20	75.46	72.99	78.32	74.77	78.61	71.96	**76.76**
Total Avg	53.69	48.58	54.58	67.74	63.95	62.45	59.80	61.58	63.73	58.14	**71.70**

5 Conclusion

In this work, we introduce *BRICK*, a new deep learning framework inspired by the biological mechanisms underlying fluctuating brain activity and cognitive state formation. Building on this foundation, we extended the concepts of neural oscillatory synchronization and attentional memory to graph domain and derive a novel graph-neural architecture, *BIG-NOS*. Extensive experiments demonstrate that *BIG-NOS* effectively alleviates over-smoothing problem in conventional GNNs, and delivers competitive performance across diverse GNN benchmarks. These results highlight the model's promise for tackling challenging graph-learning problems while offering a biologically grounded view of information propagation.

Acknowledgement

This work was supported by the National Institutes of Health (AG091653, AG068399, AG084375) and the Foundation of Hope. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the NIH.

References

[1] Juan A Acebrón, Luis L Bonilla, Conrad J Pérez Vicente, Félix Ritort, and Renato Spigler. The kuramoto model: A simple paradigm for synchronization phenomena. *Reviews of Modern Physics*, 77(1):137–185, 2005.

[2] Shun-Ichi Amari and Michael A Arbib. Competition and cooperation in neural nets. *Systems Neuroscience*, pages 119–165, 1977.

[3] Toshio Aoyagi. Network of neural oscillators for retrieving phase information. *Physical Review Letters*, 74(20):4075, 1995.

[4] Danielle S Bassett and Olaf Sporns. Network neuroscience. *Nature Neuroscience*, 20(3):353–364, 2017.

[5] Susan Y Bookheimer, David H Salat, Melissa Terpstra, Beau M Ances, Deanna M Barch, Randy L Buckner, Gregory C Burgess, Sandra W Curtiss, Mirella Diaz-Santos, Jennifer Stine Elam, et al. The lifespan human connectome project in aging: an overview. *Neuroimage*, 185:335–348, 2019.

[6] Matthew M Botvinick and Jonathan D Cohen. The computational and neural basis of cognitive control: charted territory and new frontiers. *Cognitive Science*, 38(6):1249–1285, 2014.

[7] Adrian P. Burgess. Towards a unified understanding of event-related changes in the eeg: The firefly model of synchronization through cross-frequency phase modulation. *PLOS ONE*, 7(9):e45630, 09 2012.

[8] György Buzsáki. *Rhythms of the Brain*. Oxford University Press, 10 2006.

[9] Joana Cabral, Etienne Hugues, Olaf Sporns, and Gustavo Deco. Role of local network oscillations in resting-state functional connectivity. *Neuroimage*, 57(1):130–139, 2011.

[10] Katerina Capouskova, Morten L Kringelbach, and Gustavo Deco. Modes of cognition: Evidence from metastable brain dynamics. *Neuroimage*, 260:119489, 2022.

[11] Ben Chamberlain, James Rowbottom, Maria I Gorinova, Michael Bronstein, Stefan Webb, and Emanuele Rossi. Grand: Graph neural diffusion. In *International Conference on Machine Learning*, pages 1407–1418. PMLR, 2021.

[12] Sarthak Chandra, Michelle Girvan, and Edward Ott. Continuous versus discontinuous transitions in the d-dimensional generalized kuramoto model: Odd d is different. *Physical Review. X*, 9(1), 2019.

[13] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, pages 1725–1735. PMLR, 2020.

[14] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 2018.

[15] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 257–266, 2019.

[16] Tingting Dan, Hongmin Cai, Zhuobin Huang, Paul Laurienti, Won Hwa Kim, and Guorong Wu. Neuro-rdm: an explainable neural network landscape of reaction-diffusion model for cognitive task recognition. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 365–374. Springer, 2022.

[17] Tingting Dan, Jiaqi Ding, Ziquan Wei, Shahar Kovalsky, Minjeong Kim, Won Hwa Kim, and Guorong Wu. Re-think and re-design graph neural networks in spaces of continuous graph diffusion functionals. *Advances in Neural Information Processing Systems*, 36:59375–59387, 2023.

[18] Tingting Dan, Jiaqi Ding, and Guorong Wu. Explore brain-inspired machine intelligence for connecting dots on graphs through holographic blueprint of oscillatory synchronization. *Nature Communications*, 16(1):9425, 2025.

[19] Tingting Dan, Ziquan Wei, Won Hwa Kim, and Guorong Wu. Exploring the enigma of neural dynamics through a scattering-transform mixer landscape for riemannian manifold. *International Conference on Machine Learning (ICML)*, 2024.

[20] Tahnée Engelen, Marco Solcà, and Catherine Tallon-Baudry. Interoceptive rhythms in the brain. *Nature Neuroscience*, 26(10):1670–1684, 2023.

[21] Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. *arXiv preprint arXiv:1912.09893*, 2019.

[22] Lingzhong Fan, Hai Li, Junjie Zhuo, Yu Zhang, Jiaojian Wang, Liangfu Chen, Zhengyi Yang, Congying Chu, Sangma Xie, Angela R Laird, et al. The human brainnetome atlas: a new brain atlas based on connectional architecture. *Cerebral Cortex*, 26(8):3508–3526, 2016.

[23] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *International Conference on Machine Learning*, pages 2122–2131. PMLR, 2019.

[24] Asif A. Ghazanfar and David J. Lewkowicz. The emergence of multisensory systems through perceptual narrowing. *Trends in Cognitive Sciences*, 13(11):470–478, 2009.

[25] Charles M Gray, Peter König, Andreas K Engel, and Wolf Singer. Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature*, 338(6213):334–337, 1989.

[26] Shi Gu, Fabio Pasqualetti, Matthew Cieslak, Qawi K Telesford, Alfred B Yu, Ari E Kahn, John D Medaglia, Jean M Vettel, Michael B Miller, Scott T Grafton, et al. Controllability of structural brain networks. *Nature Communications*, 6(1):8414, 2015.

[27] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1025–1035, 2017.

[28] Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Liquid time-constant networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7657–7666, 2021.

[29] SM Hadi Hosseini, Fumiko Hoeft, and Shelli R Kesler. Gat: a graph-theoretical analysis toolbox for analyzing between-group differences in large-scale structural and functional brain networks. *PloS One*, 7(7):e40709, 2012.

[30] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems*, 33:22118–22133, 2020.

[31] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106, 1962.

[32] Jamie B. Hutchinson and Nicholas B. Turk-Browne. Memory-guided attention: control from multiple memory systems. *Trends in Cognitive Sciences*, 16(12):576–579, 2012.

[33] Jeremy Kawahara, Colin J Brown, Steven P Miller, Brian G Booth, Vann Chau, Ruth E Grunau, Jill G Zwicker, and Ghassan Hamarneh. Brainnetcnn: Convolutional neural networks for brain networks; towards predicting neurodevelopment. *Neuroimage*, 146:1038–1049, 2017.

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

[35] Devin Kreuzer, Dominique Beaini, Will Hamilton, Vincent Létourneau, and Prudencio Tossou. Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems*, 34:21618–21629, 2021.

[36] Yoshiki Kuramoto. *Chemical Turbulence*, pages 111–140. Springer Berlin Heidelberg, Berlin, Heidelberg, 1984.

[37] Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[38] Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, William L. Hamilton, David Duvenaud, Raquel Urtasun, and Richard Zemel. *Efficient graph generation with graph recurrent attention networks*. Curran Associates Inc., Red Hook, NY, USA, 2019.

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

[40] Amanda L McGowan, Linden Parkes, Xiaosong He, Ovidia Stanoi, Yoona Kang, Silicia Lomax, Mia Jovanova, Peter J Mucha, Kevin N Ochsner, Emily B Falk, et al. Controllability of structural brain networks and the waxing and waning of negative affect in daily life. *Biological Psychiatry Global Open Science*, 2(4):432–439, 2022.

[41] John Dominic Medaglia, Fabio Pasqualetti, Roy H Hamilton, Sharon L Thompson-Schill, and Danielle S Bassett. Brain and cognitive reserve: Translation via network control theory. *Neuroscience & Biobehavioral Reviews*, 75:53–64, 2017.

[42] Yimeng Min, Frederik Wenkel, and Guy Wolf. Scattering gcn: Overcoming oversmoothness in graph convolutional networks. *Advances in Neural Information Processing Systems*, 33:14498–14508, 2020.

[43] Takeru Miyato, Sindy Löwe, Andreas Geiger, and Max Welling. Artificial kuramoto oscillatory neurons. *arXiv preprint arXiv:2410.13821*, 2024.

[44] Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.

[45] Vernon B Mountcastle. The columnar organization of the neocortex. *Brain: a journal of neurology*, 120(4):701–722, 1997.

[46] Tuan Nguyen, Hirotada Honda, Takashi Sano, Vinh Nguyen, Shugo Nakamura, and Tan Minh Nguyen. From coupled oscillators to graph neural networks: Reducing over-smoothing via a kuramoto model-based approach. In *International Conference on Artificial Intelligence and Statistics*, pages 2710–2718. PMLR, 2024.

[47] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*, 2020.

[48] T Konstantin Rusch, Ben Chamberlain, James Rowbottom, Siddhartha Mishra, and Michael Bronstein. Graph-coupled oscillator networks. In *International Conference on Machine Learning*, pages 18888–18909. PMLR, 2022.

[49] Daniel Senkowski, Till R Schneider, John J Foxe, and Andreas K Engel. Crossmodal binding through neural coherence: implications for multisensory processing. *Trends in Neurosciences*, 31(8):401–409, 2008.

[50] Uri Shaham, Kelly Stanton, Henri Li, Boaz Nadler, Ronen Basri, and Yuval Kluger. Spectralnet: Spectral clustering using deep neural networks. In *Proc. ICLR 2018*, 2018.

[51] Steven H Strogatz. From kuramoto to crawford: exploring the onset of synchronization in populations of coupled oscillators. *Physica D: Nonlinear Phenomena*, 143(1-4):1–20, 2000.

[52] Nathalie Tzourio-Mazoyer, Brigitte Landeau, Dimitri Papathanassiou, Fabrice Crivello, Octave Etard, Nicolas Delcroix, Bernard Mazoyer, and Marc Joliot. Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *Neuroimage*, 15(1):273–289, 2002.

[53] Nienke Van Atteveldt, Micah M Murray, Gregor Thut, and Charles E Schroeder. Multisensory integration: flexible use of general operations. *Neuron*, 81(6):1240–1253, 2014.

[54] David C Van Essen, Stephen M Smith, Deanna M Barch, Timothy EJ Behrens, Essa Yacoub, and Kamil Ugurbil. The wu-minn human connectome project: An overview. *Neuroimage*, 80:62–79, 2013.

[55] Francisco Varela, Jean-Philippe Lachaux, Eugenio Rodriguez, and Jacques Martinerie. The brainweb: phase synchronization and large-scale integration. *Nature Reviews Neuroscience*, 2(4):229–239, 2001.

[56] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

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

[58] Louis-Pascal Xhonneux, Michaël Defferrard, and Pierre Vandergheynst. Continuous graph neural networks. In *Proceedings of the 37th International Conference on Machine Learning*, pages 10432–10441. PMLR, 2020.

[59] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[60] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[61] Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. Graphformers: Gnn-nested transformers for representation learning on textual graph. *Advances in Neural Information Processing Systems*, 34:28798–28810, 2021.

[62] Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L. Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4805–4815. Curran Associates Inc., 2018.

[63] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J. Kim. Graph transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

[64] Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: current limitations and effective designs. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20. Curran Associates Inc., 2020.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to Abstract and the last paragraph of Introduction 1

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Appendix A.4

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The code is released at Anonymous GitHub.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is released at Anonymous GitHub.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Appendix A.2

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We use paired-ttest.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to Appendix A.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Appendix A.4.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please refer to Appendix A.2 for the use of data.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We release a new codebase implementing the proposed models, with documentation and instructions provided in the Anonymous GitHub.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Technical Appendices and Supplementary Material

A.1 Data Description

Table 4: Dataset description for node classification.

	Texas	Wisconsin	Actor	Squirrel	Chameleon	Cornell	Citeseer	Pubmed	Cora	ogbn-arxiv
Hom. ratio h	0.11	0.21	0.22	0.22	0.23	0.3	0.57	0.74	0.81	-
# Nodes	183	251	7,600	5,201	2,277	183	3,327	19,717	2,708	169,343
# Edges	295	466	26,752	198,493	31,421	280	4,676	44,327	5,278	1,166,243
# Classes	5	5	5	5	5	5	7	3	6	40

Table 5: Dataset description for graph classification.

	ENZYMES	PROTEINS
Avg # Nodes	32.63	39.36
Avg # Edges	62.14	39.06
# Classes	6	2
# Graphs	600	1,113

A.2 Hyperparameters

As human data as an example, for all methods, the hidden dimension is set to 256. The network depth and batch size are set to 2 and 64 for the comparison methods, while for our *BRICK*, they are set to 16 and 256, respectively.

A.3 Connections and Distinctions from Prior Approaches

A.3.1 Relevance to neuroscience domains

In principle, the learning behavior of *BRICK* is shaped by the same governing equations (i.e., Kuramoto model) that describe the neural oscillatory dynamics responsible for generating cognition and behavior. We further introduce structural-functional coupling, implemented through the geometric scattering transform, to emulate the cross-frequency coupling observed in cognitive neuroscience. In addition, we incorporate an adaptive control term to enhance task-relevant modulation. Drawing inspiration from neural oscillatory synchronization, our *BRICK* model not only improves predictive accuracy but also enhances interpretability, thereby offering novel insights into cognitive processes.

A.3.2 Advances over conventional methods in brain rhythm identification

While traditional models, including CNNs, RNNs, Transformers, and GNNs, have demonstrated strong empirical performance in brain rhythm identification, they largely remain data-driven and agnostic to the underlying neurobiological mechanisms, they often treat the brain as a generic data source without sufficient domain knowledge. In contrast, *BRICK* explicitly incorporates neural synchronization dynamics, a well-established principle in cognitive neuroscience, into the learning architecture. This alignment with neuroscience allows us to interpret learned patterns (e.g., coupling strengths or synchronization clusters) in biologically meaningful terms (as shown in Figure 3), such as inter-regional communication or pathological desynchronization. Therefore, compared to conventional approaches, *BRICK* is not just a deep model for feature representation learning, it shows the potential to establish a biologically inspired reasoning system. By embedding core principles of brain dynamics into a learnable, differentiable architecture, our proposed deep model enables achieving more interpretable, temporally-aware, and robust brain state identification.

A.3.3 Differences from prior dynamical models

While both *BRICK* and Cabral *et al.* [9] adopt the Kuramoto model to explore brain dynamics, their roles and applications diverge significantly. In their work, the Kuramoto model is used as a forward simulator to reproduce biologically observed functional connectivity (FC) patterns from empirical structural connectivity (SC), with the goal of explaining emergent brain phenomena. In contrast, our *BRICK* repurposes the Kuramoto model as a computational module within a machine learning framework. Rather than passively simulating dynamics, *BRICK* actively learns to perform representation learning, reasoning, and predictive tasks on graphs using dynamic synchronization mechanisms.

For methodological paradigm, while Capouskova *et al.* [10] follow a data-driven analytical paradigm, using autoencoders and clustering to uncover latent cognitive states from empirical fMRI data, they encode BOLD phase coherence data using modern machine learning tools without Kuramoto model. Their focus lies in neuroscientific pattern discovery and interpretation. In contrast, *BRICK* introduces a physics-informed learning framework that embeds the principles of neural synchronization, specifically via the Kuramoto model, directly into the learning process. The outcomes of these two approaches are also distinct: Capouskova *et al.*[10] generate empirical insights about cognitive brain states, whereas *BRICK* yields new machine learning models that not only excel in brain-related tasks but also generalize to broader graph-based AI problems.

Compared with [43], our *BRICK* framework offers two key advantages. First, neuroscientific grounding: starting from fMRI-derived brain rhythms, we augment the Kuramoto core with a task-driven feedback controller that mimics cognitive control, thereby preserving biological interpretability. Second, domain-specific impact: on real HCP fMRI data *BRICK* achieves the best task decoding and unsupervised parcellation, while on graph benchmarks it delivers competitive accuracy and remains resistant to oversmoothing even at 128 layers.

Taken together, these properties make *BRICK* a biologically inspired but practically efficient module that bridges brain dynamics and general-purpose AI.

A.4 Discussion and Limitation

Although our approach demonstrates promising accuracy on brain connectomes and node-level benchmarks, its computational efficiency remains limited due to the iterative solver required to integrate *BRICK* dynamics. Unlike feed-forward GNNs, our model relies on repeated geometric projections, oscillator updates, and manifold renormalization, all of which introduce substantial per-batch computational overhead. This bottleneck becomes particularly pronounced on datasets such as TUDataset.

Despite these efficiency constraints, our method still achieves competitive graph classification performance, underscoring the robustness of our dynamical-systems perspective beyond neuroscience applications. In future work, we plan to develop more efficient numerical solvers and adaptive graph batching strategies to significantly improve scalability. We also aim to apply our framework to broader multimodal and clinical datasets to further assess its generalizability and translational value.

More broadly, this line of research highlights the potential of brain-inspired synchronization mechanisms for building more interpretable and principled AI systems.

More broadly, this line of research highlights the promise of brain-inspired synchronization mechanisms for building more interpretable and efficient AI systems, while also drawing attention to important ethical considerations surrounding the use of neural data and the societal impact of autonomous decision-making technologies.