WavCIL: Wavelet Coefficient-Domain Invariant Learning for Dynamic Graph OOD Generalization

Yantong Zhu[1] **Weining Shi**[1] **Keyi Li**[1] **Xiaoyan Xie**[1] **Yi Chang**[2]
Qinggang Zhang[2,📦] **Tianjiao Pu**[3] **Ji Qiao**[3] **Zhihong Zhang**[1,†]

[1]Xiamen University, Xiamen, China
[2]Jilin University, Changchun, China
[3]China Electric Power Research Institute Co., Ltd., Beijing, China
{zhuyantong, weining, likeyi, xiexiaoyan}@stu.xmu.edu.cn
{yichang, qinggangzhang}@jlu.edu.cn
{tjpu, qiaoji}@epri.sgcc.com.cn, zhihong@xmu.edu.cn

Abstract

Dynamic Graph Neural Networks (DyGNNs) demonstrate powerful representation capabilities in crucial time-series systems by leveraging graph structure and temporal dynamics. However, existing dynamic graph models often suffer from unpredictable distribution shifts induced by complex time-varying factors such as emergency incidents, leading to severe performance degradation due to their low out-of-distribution (OOD) generalization. Graph invariant learning has been extensively studied to solve this problem through feature disentanglement in the time or spectral domain. Despite recent advances, existing methods suffer from critical limitations in practice: Temporal-based methods struggle to disentangle transient perturbations and stable patterns that are highly overlapping and entangled within the same period. While spectral-based methods attempt spectral-domain disentanglement via the global Fourier transform, these methods inherently lose temporal localization and fail to effectively characterize distribution shifts driven by time-varying factors. To this end, in this paper, we propose **Wav**elet **C**oefficient-Domain **I**nvariant **L**earning **(WavCIL)**, a novel framework for resolving distribution shifts by conducting disentanglement within a wavelet coefficient domain endowed with time-frequency localization capabilities. Specifically, WavCIL consists of two key components: (i) a wavelet transform module, which leverages a set of wavelet bases to map the input temporal signals into wavelet coefficient domain, simultaneously capturing frequency components and localizing their temporal occurrences; and (ii) a coefficient domain invariant learning module, which utilizes an adaptive mask mechanism to disentangle stable causal patterns invariant to time-varying factors from spurious patterns driven by time-varying factors in the coefficient domain, guiding the prediction process to rely more on the causal patterns. Extensive experiments on several benchmark datasets demonstrate that WavCIL achieves state-of-the-art generalization performance when handling distribution shifts. Our code is released at https://github.com/yantongzhu/WavCIL.

1 Introduction

Dynamic Graph Neural Networks (DyGNNs) integrate topological structure and temporal evolutionary information, thereby demonstrating strong representation capabilities in crucial time-series

†Corresponding author. 📦 Project Leader.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

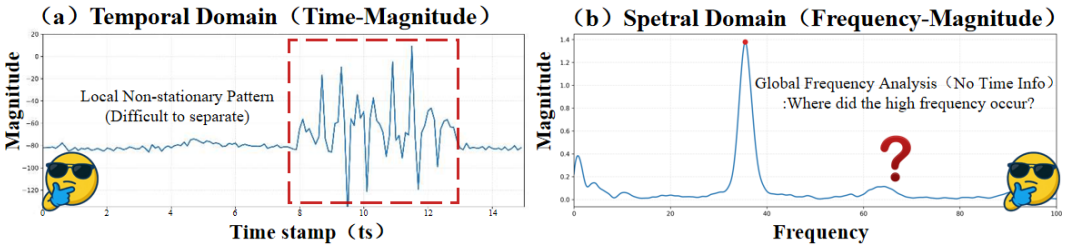

Figure 1: Comparison of invariant learning paradigms. (a) Temporal-based methods struggle with deeply entangled factors. (b) Spectral-based methods via Fourier transform lose temporal localization.

systems such as social network evolution analysis and financial transaction risk forecasting[1, 2, 3]. In the real world, dynamic graphs are always influenced by multiple latent and time-varying factors, including unforeseen emergency incidents, policy cycles, or seasonal trends[4, 5], which often lead to unpredictable temporal distribution shifts[1, 4, 6]. Such shifts violate the independent and identically distributed (I.I.D.) assumption central to classical machine learning, causing the patterns that models learn from historical data to become ineffective. This leads to critical performance degradation and severe decision-making risks due to insufficient out-of-distribution (OOD) generalization ability[6, 7], the enhancement of which has become a critical challenge for reliable real-world applications[5, 6].

Recently, graph invariant learning has been extensively studied to improve OOD generalization by leveraging data from diverse distributions to learn stable causal patterns invariant to time-varying factors[7]. Generally, existing work can be broadly categorized into two main directions: **(i)** Temporal-based Invariant Learning, which identifies invariant patterns directly in the time domain by designing disentangled spatio-temporal attention or environment inference mechanisms[1, 4]; **(ii)** Spectral-based Invariant Learning, which transforms embeddings into the spectral domain and employs spectral analysis tools such as Fourier transform to disentangle causal and spurious patterns[5, 8].

Despite their effectiveness, existing approaches suffer from critical limitations in real-world scenarios. **(i)** Temporal-based methods rely on an assumption that the time-varying factors causing distribution shifts are separable or observable in the time domain. However, this assumption is easily violated in real-world scenarios. When the influence of time-varying factors is deeply entangled with causal mechanisms in time domain, as illustrated in Figure 1(a), such methods struggle to disentangle causal mechanisms from time-varying factors[1, 4]. **(ii)** Although spectral-based methods provide a new perspective for disentanglement by Fourier transform, this global analysis tool loses the localization information of frequency components along the time axis, as shown in Figure 1(b). Therefore, it fails to characterize non-stationary distribution shifts driven by local events or phased variations[5, 8]. Take a real-time MOOC system as an example where students and courses act as nodes and homework submissions serve as edges: When long-term submission patterns (causal patterns invariant to time-varying factors) and pre-deadline bursts (distribution shifts induced by external factors) overlap within the same time period, temporal-based methods fail to disentangle them. While spectral-based methods lack temporal localization, they cannot determine whether extracted high-frequency features originate from a long-term intensive submission habit or a transient sharp increase before a deadline.

In this paper, we propose to study distribution shifts on dynamic graphs in the wavelet coefficient domain. As illustrated in Figure 2, this domain provides a powerful representation space, where highly complex distribution shifts induced by time-varying factors can be characterized as localized non-stationary patterns. However, effectively modeling such distribution shifts is still challenging: **(a)** It is hard to accurately characterize frequency components without losing their temporal localization when transforming graph node features sequences into the coefficient domain, which yields time-frequency representations. **(b)** It is non-trivial to effectively disentangle causal patterns from spurious patterns, which remain deeply entangled even with time-frequency representations.

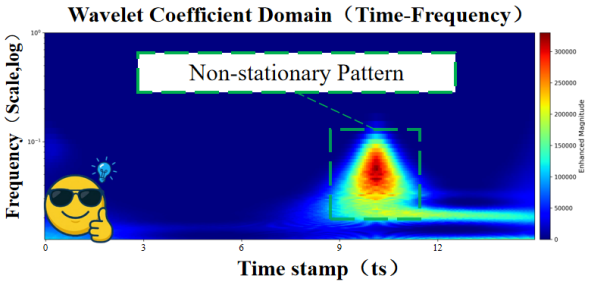

Figure 2: Characterizing distribution shifts as localized non-stationary patterns in wavelet coefficient domain.

To this end, we propose **Wavelet Coefficient-Domain Invariant Learning (WavCIL)**, a novel framework for explicitly resolving distribution shifts by conducting fine-grained disentanglement within a wavelet coefficient domain endowed with time-frequency localization capabilities. WavCIL comprises two innovative modules: **(a)** The Wavelet Transform Module, which constructs a coefficient-domain representation from input time-series signals via a set of translatable and scalable wavelet bases. This representation reflects both frequency components and their temporal localization. **(b)** The Coefficient-Domain Invariant Learning Module, which designs adaptive masks within this domain to directly disentangle stable causal patterns invariant to time-varying factors and spurious patterns driven by time-varying factors with respect to label prediction. It then optimizes an invariant learning objective to ensure predictions rely solely on the causal patterns, thereby effectively eliminating the influence of spurious patterns. Generally, our contributions are summarized as follows:

- We identify the key limitation of existing graph invariant learning methods and propose WavCIL to achieve strong OOD generalization without sacrificing time-frequency local information.
- WavCIL adopts a novel Wavelet Transform Module to map node feature sequences into the wavelet coefficient domain, and designs a Coefficient-Domain Invariant Learning Module, allowing the model to disentangle causal patterns from spurious patterns within this domain.
- Extensive experiments on multiple synthetic and real-world dynamic graph datasets demonstrate that our WavCIL method consistently and significantly outperforms existing representative baselines in OOD generalization performance, validating the empirical superiority of WavCIL.

2 Problem Statement

In this section, we formally define the notations for dynamic graphs and the invariant learning framework for OOD generalization under complex distribution shifts caused by time-varying factors.

Let $\mathcal{G} = \{\mathrm{G}^1, \mathrm{G}^2, \ldots, \mathrm{G}^T\}$ denote a discrete-time dynamic graph over a time window of length T, where $\mathrm{G}^t = (\mathrm{V}, \mathrm{E}^t, \mathrm{X}^t)$ represents the graph snapshot at time step t. Here, V denotes a fixed set of nodes with $N = |\mathrm{V}|$, E^t is the set of edges, and $\mathbf{X}^t \in \mathbb{R}^{N \times d_0}$ represents the initial node attributes. Given the historical sequence, the core objective of dynamic graph representation learning is to learn a low-dimensional and dense vector representation for the feature sequence X of a given target, and to use this representation to infer the true label Y of the target at a future time step[2, 9].

In real-world scenarios, the evolution of dynamic graphs is driven by multiple latent time-varying factors, leading to unpredictable distribution shifts[1]. To achieve OOD generalization against such shifts, we formalize invariant learning for dynamic graphs into the following three core stages:

(i) **Spatio-temporal Representation Learning.** Given a historical feature sequence X, the model first employs a GNN **enc**oder $f_{enc}(\cdot)$ to extract a foundational temporal feature trajectory $\mathbf{Z} = f_{enc}(X)$. Under the standard ERM paradigm, the extracted $\mathbf{Z}$ inevitably entangles stable causal patterns invariant to time-varying factors with spurious patterns driven by specific time-varying factors.

(ii) **Invariant Feature Disentanglement.** To overcome distribution shifts, the model must disentangle the entangled features via a feature **dis**entanglement mask $f_{dis}(\cdot)$, i.e., $[\mathbf{Z}_c, \mathbf{Z}_s] = f_{dis}(\mathbf{Z})$, thereby extracting **c**ausal representations $\mathbf{Z}_c$ and **s**purious representations $\mathbf{Z}_s$. In this paper, to accurately capture non-stationary variations driven by local events, we focus on modeling the disentanglement mask $f_{dis}(\cdot)$ in the wavelet coefficient domain, which has time-frequency localization properties.

(iii) **Precisely Prediction Generation.** Finally, the inference process for the downstream target label Y is strictly constrained to depend only on the disentangled causal representations $\mathbf{Z}_c$, i.e., the **pred**icted probability is given by $P(Y \mid X) = f_{pred}(\mathbf{Z}_c)$. The essence of this framework is to break the dependency of the prediction on spurious representations $\mathbf{Z}_s$. Simply minimizing the empirical risk based on the causal representation is equivalent to minimizing the worst-case risk across all potential distribution shifts, thereby ensuring reliable inference in completely unseen open scenarios.

$$\min_{\theta} \mathbb{E}_{(X,Y)}[\mathcal{L}(f_{pred}(\mathbf{Z}_c), Y)] \tag{1}$$

where θ denotes the learnable parameters of the model and $\mathcal{L}(\cdot, \cdot)$ is the task-specific loss function.

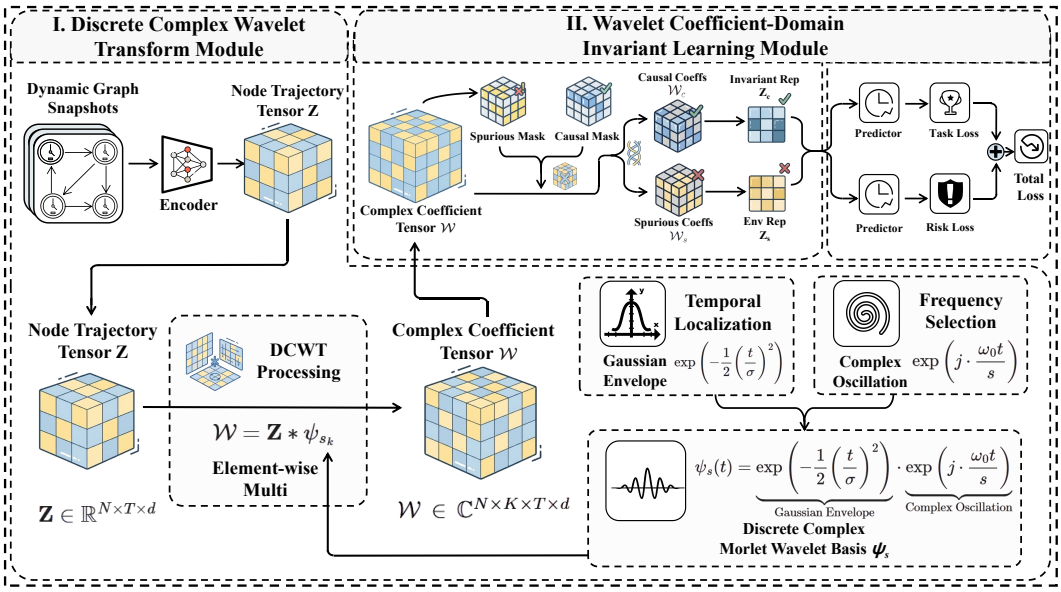

Figure 3: **The Overall Framework of WavCIL.** WavCIL consists of two key components: (i) Discrete Complex Wavelet Transform, which projects node temporal feature sequences into a multi-scale wavelet coefficient domain; and (ii) Wavelet Coefficient-Domain Invariant Learning, which leverages an adaptive dual mask mechanism to disentangle causal and spurious patterns, and an intervention-based variance penalty is jointly optimized to ensure OOD generalization.

3 The Framework of WavCIL

In this section, we introduce **WavCIL** in detail. To improve OOD generalization, WavCIL transforms representations into wavelet coefficient domain with time–frequency localization properties and thereby disentangle within this domain. The whole framework is illustrated in Figure 3.

3.1 Fundamentals of Discrete Complex Morlet Wavelet Theory

To analyze the evolution of frequency over time in dynamic graphs, it is necessary to employ complex analytic wavelets to separate amplitude and phase. Mathematically, for a function $\psi \in L^2(\mathbb{R})$ to be an analytic wavelet, its Fourier transform $\hat{\psi}(\omega)$ must be strictly zero at negative frequencies[10]:

$$\hat{\psi}(\omega) = 0, \quad \text{if } \omega < 0. \tag{2}$$

To construct time-frequency atoms satisfying this condition, this is achieved by frequency modulation of a real-valued and symmetric low-pass window function $g(t)$, i.e., $\psi(t) = g(t)\exp(j\eta t)$. Here, t denotes the time variable, j is the imaginary unit, and η represents the specified modulation center frequency. By the frequency-shift property of the Fourier transform, its frequency-domain representation is a shift of the spectrum of the original window function: $\hat{\psi}(\omega) = \hat{g}(\omega - \eta)$.

To achieve the theoretical optimal lower bound of the Heisenberg Uncertainty Principle between time localization and frequency resolution, we select the Gaussian function as the window function, given by $g(t) \propto \exp(-t^2/2\sigma^2)$, where σ is the standard deviation controlling the window width, thereby constructing the Morlet wavelet[11]. As long as the condition $\sigma^2\eta^2 \gg 1$ is satisfied, i.e., the product of the bandwidth and the center frequency is sufficiently large, the negative-frequency component of its Fourier transform $\hat{g}(\omega)$ can be approximated as zero, thus becoming a strictly analytic wavelet.

Building upon the aforementioned theory, we introduce the Discrete Complex Morlet Wavelet. It acts as a mathematical "microscope" capable of adjusting its focus in time and frequency simultaneously. Formally, for a specific scale s and time t, the wavelet bases $\psi_s(t)$ are defined as follows:

$$\psi_s(t) = \underbrace{\exp\left(-\frac{1}{2}\left(\frac{t}{\sigma}\right)^2\right)}_{\text{Gaussian Envelope}} \cdot \underbrace{\exp\left(j \cdot \frac{\omega_0 t}{s}\right)}_{\text{Complex Oscillation}} \tag{3}$$

Derivation is detailed in Appendix B.1

where σ is the bandwidth parameter, t denotes the centered time index, and ω_0 is the center frequency parameter. This formulation consists of two collaboratively operating components that jointly aim to effectively tackle the distribution shift problem across diverse unknown dynamic test scenarios:

- **Gaussian Envelope (Temporal Localization):** The exponential decay term $\exp(-t^2/2\sigma^2)$ constructs a bell-shaped window centered at t, which is equivalent to the real-valued symmetric window function of the Gabor wavelet. This envelope guarantees that the extracted coefficients are only sensitive to signal variations within the local temporal neighborhood of the window.
- **Complex Oscillation (Frequency/Scale Selection):** The complex exponential term $\exp(j \cdot \omega_0 t/s)$ defines a rotating vector in the complex plane. When the input trajectory contains frequency components corresponding to scale s, this term resonates with the signal, thereby extracting specific periodic patterns such as periodic activity patterns of users.

The detailed mathematical derivation regarding time-frequency atoms, Heisenberg boxes for optimal localization, and the evolution from continuous to discrete transforms is provided in Appendix B.1.

3.2 Discrete Complex Wavelet Transform

Building on the aforementioned wavelet theoretical foundation, this subsection introduces how the Discrete Complex Wavelet Transform (DCWT) module encodes the original dynamic graph snapshots into continuous node trajectories and maps them into a multi-scale coefficient domain.

Given a dynamic graph sequence containing N nodes $\mathcal{G} = \{\mathrm{G}^1, \mathrm{G}^2, \ldots, \mathrm{G}^T\}$, where $\mathrm{G}^t = (\mathrm{V}, \mathrm{E}^t, \mathrm{X}^t)$ denotes the graph snapshot at time step t. To maintain the generality of the framework, we adopt the standard Graph Message Passing Paradigm as the spatial encoder. For any node $v \in \mathrm{V}$, the update process of its feature representation $\mathbf{h}_v^{t,(l)}$ at time snapshot t and layer l is formalized as:

$$\mathbf{h}_v^{t,(l)} = \mathrm{UPDATE}^{(l)}\left(\mathbf{h}_v^{t,(l-1)},\ \mathrm{AGG}^{(l)}\left(\left\{\mathrm{MSG}^{(l)}\left(\mathbf{h}_v^{t,(l-1)}, \mathbf{h}_u^{t,(l-1)}\right) : u \in \mathcal{N}^t(v)\right\}\right)\right). \tag{4}$$

where $\mathcal{N}^t(v)$ denotes the set of neighbors of node v at time t; $\mathbf{h}_v^{t,(0)} = \mathbf{X}_v^t$ represents the initial node attributes; $\mathrm{MSG}^{(l)}(\cdot)$ is the message computation function; and $\mathrm{AGG}^{(l)}(\cdot)$ denotes the neighborhood aggregation operation. By stacking multiple layers of this operation, we can extract a compact and structural representation $\mathbf{h}_v^t \in \mathbb{R}^d$ for each node under the static graph snapshot at that time step.

By aggregating the outputs across all time steps, we construct the temporal feature trajectory $\mathbf{Z}_v = [\mathbf{h}_v^1, \mathbf{h}_v^2, \ldots, \mathbf{h}_v^T] \in \mathbb{R}^{T\times d}$ for node v. On this basis, we construct the global-graph time-series feature trajectory tensor $\mathbf{Z} = \{\mathbf{Z}_v \mid v \in \mathrm{V}\} \in \mathbb{R}^{N\times T\times d}$. This trajectory serves as the original signal, encapsulating the entangled stationary structures and non-stationary evolution patterns[12].

Based on the wavelet bases $\{\psi_{s_k}\}_{k=1}^K$, we project the trajectory $\mathbf{Z}$ onto the wavelet coefficient domain. For each feature dimension, we compute the convolution between the trajectory and the bases:

$$\mathcal{W} = \mathbf{Z} * \psi_{s_k}, \quad k \in \{1, \ldots, K\}. \tag{5}$$

Finally, we obtain the complex wavelet coefficient tensor $\mathcal{W} \in \mathbb{C}^{N\times K\times T\times d}$. Each element in this tensor characterizes the evolutionary features of node v at time step t and scale s_k, preserving the locally time-varying characteristics and multi-scale frequency structure of node evolution. By simultaneously characterizing temporal locality and frequency information, the DCWT provides coefficient representations that serve as a information foundation for follow-up fine-grained disentanglement.

3.3 Wavelet Coefficient-Domain Invariant Learning

Adaptive dual mask. After obtaining the multi-scale representation $\mathcal{W}$, we propose a Wavelet Coefficient-Domain Invariant Learning Module to achieve disentanglement of causal and spurious patterns via adaptive dual masking. We first concatenate the real and imaginary parts of the complex coefficients within the observation window into $\mathbf{H}_{in} \in \mathbb{R}^{N\times(2Kd)}$, and then employ a parameterized mapping network Φ_{mask} to generate latent importance logits $\mathbf{L} = \Phi_{mask}(\mathbf{H}_{in}) \in \mathbb{R}^{N\times K}$. To achieve strict disentanglement, we employ a Sigmoid function with temperature scaling to generate two complementary masks, the causal mask $\mathbf{M}_c$ and the spurious mask $\mathbf{M}_s$ for subsequent filtering:

$$\mathbf{M}_c = \sigma\left(\frac{\mathbf{L}}{\tau}\right), \quad \mathbf{M}_s = \sigma\left(-\frac{\mathbf{L}}{\tau}\right) = 1 - \mathbf{M}_c, \tag{6}$$

where $\sigma(\cdot)$ denotes the Sigmoid function and $\tau(\tau > 0)$ is the temperature hyperparameter. The design principles of the complementary constraint for dual masks are detailed in Appendix B.3.

The generated masks serve as adaptive filters applied to the complex wavelet coefficient domain. We separate the original coefficients into causal and spurious components via element-wise multiplication:

$$\mathcal{W}_c = \mathcal{W} \odot \mathbf{M}_c, \quad \mathcal{W}_s = \mathcal{W} \odot \mathbf{M}_s, \tag{7}$$

where $\mathcal{W}_c, \mathcal{W}_s \in \mathbb{C}^{N\times K\times d}$. Then, we adopt a linear projection layer Φ_{proj} to flatten and compress the real and imaginary parts of the filtered coefficients, mapping them back to the real-valued feature space respectively to derive the causal representation $\mathbf{Z}_c$ and the spurious representation $\mathbf{Z}_s$.

Optimization objective. To guide the learning of masks and ensure the OOD generalization capability of $\mathbf{Z}_c$, we employ a hybrid optimization objective that combines prediction accuracy with environmental invariance constraints. The primary objective is to ensure that the causal representation contains sufficient information to accomplish the downstream task under unknown distribution shifts.

We adopt a predictor to estimate the target label Y based on $\mathbf{Z}_c$, minimizing the standard prediction loss $\mathcal{L}_{task}$. Additionally, to theoretically guarantee that the extracted representation $\mathbf{Z}_c$ contains no spurious environment information, we introduce a variance regularization based on intervention.

We define the environmental loss $\mathcal{L}_{env}$ as the sum of the mean and variance of the prediction loss under intervention. The variance term computes the variance of the loss across multiple random interventions. Minimizing this variance forces the model to break the dependency between predictions and $\mathbf{Z}_s$, driving the mask generator to assign unstable, environment-sensitive coefficients to $\mathcal{W}_s$. The definition is detailed in Appendix B.3. The final overall optimization objective is given by:

$$\mathcal{L}_{total} = \underbrace{\mathcal{L}_{task}}_{\text{Eq. (35)}} + \lambda \cdot \underbrace{\mathcal{L}_{env}}_{\text{Eq. (36)}}. \tag{8}$$

By minimizing this overall objective, WavCIL can automatically suppress spurious correlations deeply coupled with specific contexts during training, thus maintaining precise prediction performance when facing unknown distribution shifts during the subsequent testing phase. Ultimately, WavCIL disentangles causal patterns from spurious patterns and enforces the model to rely exclusively on the causal patterns, thereby ensuring strong out-of-distribution generalization under distribution shifts.

4 Experiment

In this section, we conduct extensive experiments to comprehensively evaluate the effectiveness of the proposed **WavCIL**. Our experimental design aims to address the following three key questions: **Q1 (Effectiveness)**: Faced with the distribution shifts caused by natural evolution in real-world, does WavCIL achieve superior performance over existing strong baseline methods on both link prediction and node classification tasks? **Q2 (OOD Generalization under Extreme Shifts)**: Under controlled and progressively increasing interference of synthesized spurious correlations, can WavCIL consistently maintain stable prediction performance? **Q3 (Ablation Study)**: What specific contributions do each key component of WavCIL make to the OOD generalization performance of the model?

4.1 Experimental Setups

Datasets. We evaluate WavCIL on two core tasks, link prediction and node classification, over 8 dynamic graph datasets. Task definitions and evaluation metrics are detailed in Appendix D.1 & D.2.

- **Real-world Datasets.** We employ 6 real-world dynamic graph datasets. For link prediction task (Wiki[13], Flight[14], Reddit[13], MOOC[13]), the model is required to predict whether new edges will form between nodes at the next time step, using historical dynamic graph snapshots. We partition the historical dynamic graph snapshots along the time axis to evaluate model performance under distribution shifts caused by the natural evolution of real-world systems. For node classification tasks (Aminer[15, 16], OGBN-Arxiv[17, 18]), we adopt a strict year-based partitioning strategy, requiring the model to overcome distribution shifts caused by evolution of research preferences and citation patterns across different periods to make accurate predictions.
- **Synthetic Datasets.** To quantitatively evaluate the model's ability to cope with distribution shifts of varying intensities, we construct synthetic datasets with controllable shift levels (i.e., 0.4, 0.6, 0.8)[5]. During training, we manually inject controllable spurious correlations into the data for link prediction (based on Collab[19, 1]) and node classification (based on the Stochastic Block Model, SBM). During testing, all shift levels are set to 0 to verify whether the model has truly learned cross-environment invariant causal patterns for precise predictions in unknown factors.

Baselines. We compare the performance of WavCIL with three categories of strong baseline methods:

- **General Dynamic Graph Models**: including DyGFormer[20], GraphMixer[21], DEFT-T[22] and TGN[23]. These models have strong spatio-temporal modeling capabilities but are typically built upon the independent and identically distributed assumption, ignoring distribution shifts.
- **Generic OOD Methods**: including IRM[24], V-REx[25], and GroupDRO[26]. We apply these classic out-of-distribution optimization strategies to dynamic graph backbones as baselines.
- **Dynamic Graph OOD Methods**: including DIDA[1], SILD[5], EAGLE[4], and STONE[27]. These models represent the current approaches for handling distribution shifts in dynamic graphs, either in the spatio-temporal or spectral domain ensuring highly generalization performance.

Further dataset and baseline details are provided in Appendix D.3 & Appendix D.4.

4.2 Effectiveness (Q1)

To address **Q1**, we comprehensively compare the performance of our proposed WavCIL against various representative baseline methods on diverse real-world datasets. Table 1 and Table 2 respectively summarize the detailed experimental results for both link prediction and node classification.

Table 1: Detailed performance comparison on real-world and synthetic dynamic link prediction tasks (AUC %). The best results are clearly highlighted in bold, and the second-best are underlined.

	Real-world Datasets (AUC %)					Synthetic Datasets (AUC %)				Overall (AUC %)	
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**Average**	**0.4**	**0.6**	**0.8**	**Average**	**Average**	**Δ**
General Dynamic Graph Models											
DyGFormer	83.16±0.39	84.66±0.09	**96.04±0.06**	57.03±0.92	80.22	86.94±0.27	87.26±0.29	87.09±0.03	87.10	83.17	+5.66
GraphMixer	64.78±1.15	78.79±0.65	90.13±0.15	57.49±0.37	72.80	83.34±0.56	82.97±0.84	82.63±0.84	82.98	77.16	+11.67
TGN	75.50±0.93	88.04±0.42	77.24±0.34	54.65±0.72	73.86	85.08±0.61	84.53±0.34	83.84±0.28	84.48	78.41	+10.42
DEFT-T	54.01±1.26	61.73±4.60	78.88±13.11	63.75±6.32	64.59	73.25±3.47	60.77±3.91	56.06±6.07	63.36	64.06	+24.77
Generic OOD Methods											
IRM	82.98±0.14	66.61±0.42	90.39±0.45	63.10±1.18	75.77	56.83±0.21	59.55±0.45	59.03±0.35	58.47	68.36	+20.47
VREx	83.34±0.09	64.85±0.75	91.27±0.30	63.77±0.42	75.81	76.61±1.71	75.18±0.21	74.15±0.14	75.31	75.60	+13.23
GroupDRO	84.17±0.36	65.65±2.23	94.86±0.25	64.33±0.98	77.25	75.67±1.59	73.70±0.73	73.63±0.33	74.33	76.00	+12.83
Dynamic Graph OOD Methods											
STONE	84.80±0.23	95.91±0.05	88.63±0.34	55.51±0.22	81.21	71.77±0.15	68.44±0.09	68.14±0.06	69.45	76.17	+12.66
DIDA	84.94±0.12	62.67±0.03	93.40±0.43	63.98±2.71	76.25	85.20±0.84	82.89±0.23	72.59±3.31	80.23	77.95	+10.88
SILD	80.02±0.60	91.98±1.26	86.73±1.68	68.97±10.00	81.93	85.96±0.19	84.64±1.25	78.01±0.70	82.87	82.33	+6.50
EAGLE	81.87±0.13	**97.63±0.02**	78.96±3.10	61.51±0.05	79.99	88.32±0.61	87.29±0.71	82.30±0.75	85.97	82.55	+6.28
Our Method											
WavCIL (Ours)	**85.76±0.64**	97.13±0.09	90.04±0.70	**71.26±4.71**	**86.05**	**90.87±0.33**	**90.72±0.23**	**91.01±0.11**	**90.87**	**88.83**	-

Obs. 1: WavCIL achieves superior performance on real-world link prediction tasks. Table 1 summarizes the results of link prediction. WavCIL achieves the highest overall average AUC (86.05%) across all datasets. Notably, on datasets such as Wiki, MOOC, and Reddit, which are influenced

by localized events, WavCIL significantly outperforms the spectral-domain baseline SILD. This demonstrates that, in contrast to global Fourier transform, which loses temporal locality entirely, local time–frequency disentanglement via Discrete Complex Wavelet Transform (DCWT) enables more precise capture of non-stationary causal patterns driven by local events or phased changes.

Table 2: Detailed performance comparison on real-world and synthetic temporal node classification tasks (ACC %). The best results are clearly highlighted in bold, and the second-best are underlined.

	Real-world Datasets (ACC %)					Synthetic Datasets (ACC %)				Overall (ACC %)	
Model	**Aminer15**	**Aminer16**	**Aminer17**	**OGBN-Arxiv**	**Average**	**0.4**	**0.6**	**0.8**	**Average**	**Average**	**Δ**
General Dynamic Graph Models											
DyGFormer	16.43±0.86	17.98±0.94	17.40±0.84	26.90±0.34	19.68	22.55±0.01	23.20±0.02	24.14±0.02	23.30	21.23	+26.99
GraphMixer	13.55±2.70	15.53±1.98	14.43±2.81	26.39±0.13	17.48	22.62±0.18	23.36±0.21	24.16±0.09	23.38	20.01	+28.21
TGN	11.71±0.28	13.74±0.29	12.16±0.30	12.54±1.29	12.54	24.96±0.99	25.84±0.98	25.91±1.42	25.57	18.12	+30.10
DEFT-T	44.20±5.14	46.21±4.32	36.85±4.66	14.99±0.07	35.56	43.33±2.11	40.00±4.16	40.90±5.37	41.41	38.07	+10.15
Generic OOD Methods											
IRM	48.44±0.13	50.18±0.73	42.40±0.27	51.00±0.52	48.00	33.23±4.70	30.29±1.71	29.43±1.38	30.98	40.71	+7.51
VREx	48.70±0.73	49.24±0.27	42.59±0.37	52.60±0.45	48.28	41.78±1.30	38.11±2.81	29.56±0.44	36.48	43.23	+4.99
GroupDRO	48.73±0.61	49.74±0.26	42.80±0.36	55.25±0.09	49.13	41.35±2.19	35.74±3.93	31.03±1.24	36.04	43.52	+4.70
Dynamic Graph OOD Methods											
STONE	46.20±1.78	50.85±0.89	41.98±1.30	50.00±1.52	47.26	24.89±1.59	24.38±0.34	28.54±0.62	25.94	38.12	+10.10
DIDA	50.34±0.81	51.43±0.27	44.69±0.06	50.41±0.45	49.22	43.33±7.74	39.48±7.93	28.14±3.07	36.98	43.97	+4.24
SILD	51.06±0.80	54.00±1.16	44.77±0.43	53.36±3.65	50.80	43.82±3.97	40.95±2.64	38.88±2.00	41.22	46.69	+1.53
EAGLE	51.34±0.68	**54.08±0.78**	45.53±0.47	57.90±4.25	52.21	26.92±0.31	28.33±0.91	31.48±0.23	28.91	42.23	+5.99
Our Method											
WavCIL (Ours)	**51.88±0.76**	53.78±0.41	**45.57±1.01**	**58.53±0.68**	**52.44**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**	**42.59**	**48.22**	-

Obs. 2: WavCIL performs excellently on real-world node classification tasks. Table 2 reports the classification accuracy (ACC) for node classification. WavCIL consistently outperforms all baseline methods on overall average accuracy across all datasets (52.44%). On the highly challenging Aminer and OGBN-Arxiv datasets, where academic citation preferences and patterns shift significantly over years, WavCIL still maintains better performance than EAGLE and SILD. This confirms that isolating causal relationships in the wavelet coefficient domain provides more accurate representations for the evolution of node semantics across long time spans, ensuring strong OOD generalization capabilities.

4.3 OOD Generalization Under Extreme Shifts (Q2)

To address **Q2**, we evaluate the generalization capability of the models under extreme and controllable distribution shifts. We analyze the performance of each model on the Synthetic datasets under varying levels of spurious correlation intensity (shift levels of 0.4, 0.6, and 0.8), as illustrated in Figure 4.

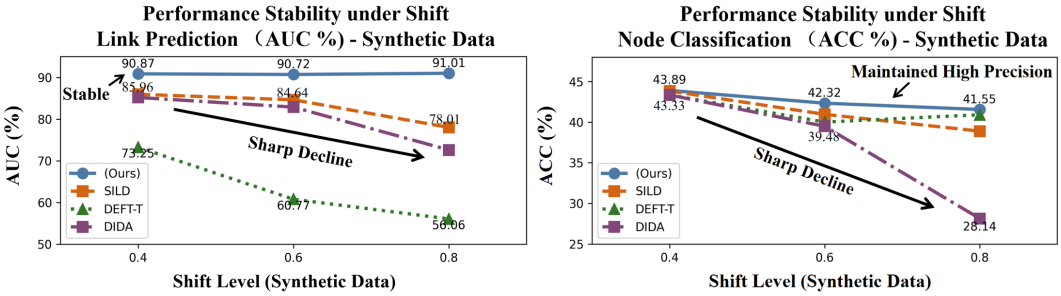

Figure 4: Comparison of the performance stability of each model under different distribution shift intensities. Evaluated on synthetic datasets for link prediction (left) and node classification (right).

Obs. 3: WavCIL demonstrates superior anti-interference capability and stability against extreme distribution shifts. As shown in Figure 4, experimental results show that the performance of most baseline models suffers a sharp decline as the shift level increases, meaning the spurious correlations in the training phase become stronger and more deceptive. For example, under high-intensity shifts, models easily misidentify highly time-varying spurious patterns as predictive signals. The AUC of SILD show a rapid fall from 85.96% at a shift level of 0.4 to 78.01% at 0.8. In contrast, WavCIL maintains extremely high AUC scores of 90.87%, 90.72%, and 91.01% at shift levels of 0.4, 0.6, and 0.8 respectively. Whether on link prediction or node classification tasks, WavCIL consistently sustains high predictive accuracy with minimal performance degradation. These results validate that the wavelet-based localized representations and coefficient-domain disentanglement mechanism

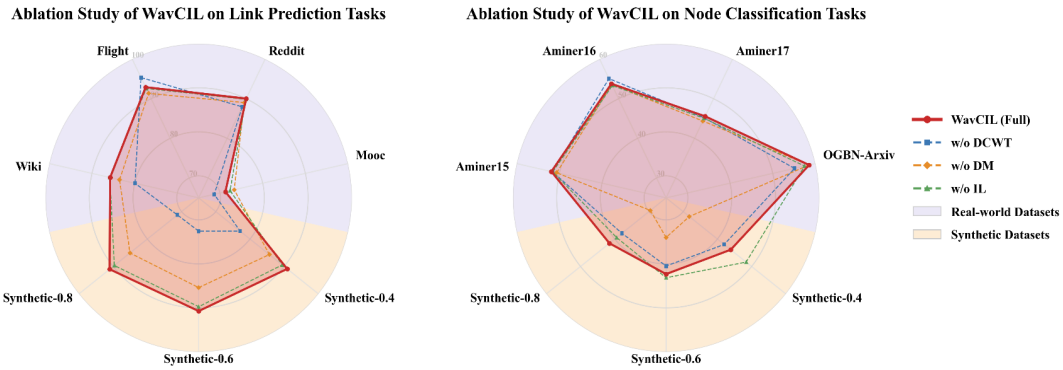

Figure 5: Comprehensive ablation study of core WavCIL components. Evaluated rigorously on link prediction (left) and node classification (right) tasks across diverse real-world and synthetic datasets.

in WavCIL successfully break the model's reliance on spurious correlations coupled with specific contexts, ensuring highly strong OOD generalization performance across diverse environments.

4.4 Ablation Study (Q3)

To address **Q3** and investigate the individual contributions of core component in the WavCIL framework to OOD generalization performance, we conduct a comparative analysis between full model and three degraded variants to comprehensively demonstrate the necessity of our proposed designs, as illustrated in Figure 5. The specific configurations of these variants are described as follows:

- **w/o DCWT** (without Discrete Complex Wavelet Transform): In the frequency-domain filtering stage, the global Fast Fourier Transform (FFT) is used to replace the Discrete Complex Morlet Wavelet Transform (DCWT), resulting in a complete loss of crucial temporal locality.
- **w/o DM** (without Disentanglement Mask): The adaptive mask learning module in the coefficient domain is disabled, and all wavelet coefficient-domain signals are directly fed into downstream prediction, preventing the forced separation of causal patterns and spurious patterns.
- **w/o IL** (without Invariant Learning): The environment invariance penalty term is removed from the final optimization objective (i.e., the intervention penalty coefficient is set to $\lambda = 0$). The model thus degenerates to relying solely on Empirical Risk Minimization (ERM) for training.

Obs. 4: Each core component of the WavCIL framework contributes significantly to OOD generalization. Compared with the full model, all three degraded variants suffer from performance degradation to varying degrees across all evaluation metrics, as shown in Figure 5. These results confirm that: when handling non-stationary distribution shifts in dynamic graphs, the wavelet coefficient domain representations with time–frequency localization form the fundamental basis for capturing dynamic evolutionary patterns; the adaptive mask separation mechanism provides a critical guarantee for effectively removing time-varying factors; and the environment invariance optimization is a necessary condition for enhancing the model's generalization capability across environments. Full quantitative results and performance degradation analysis for ablation variants on the full dataset are provided in Appendix C.1. More additional experiments are provided in Appendix C.

5 Conclusion

DyGNNs often suffer from severe performance degradation under distribution shifts. While graph invariant learning has shown promise in enhancing model OOD generalization capabilities through cross-environment feature disentanglement, addressing complex distribution shifts remains a significant challenge. This paper identifies and analyzes a critical bottleneck in current invariant learning methods: existing spectral-domain approaches attempt to extract causal patterns via the global Fourier transform, but inherently suffer from a complete loss of temporal localization. To address these challenges, we propose the novel WavCIL that resolves distribution shifts within a wavelet coefficient domain endowed with time-frequency localization capabilities. Specifically, WavCIL first projects node temporal trajectories into a multi-scale wavelet coefficient domain, and subsequently disen-

tangles causal and spurious patterns using an adaptive dual-mask mechanism to resolve the shifts. Experiments show that WavCIL outperforms the representative competitors under distribution shifts.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62676337, 52307153) and the Xiamen Major Science and Technology Project (No. 3502Z20241028).

References

[1] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Zhou Qin, and Wenwu Zhu. Dynamic graph neural networks under spatio-temporal distribution shift. *Advances in neural information processing systems*, 35:6074–6089, 2022.

[2] Pengfei Jiao, Hongjiang Chen, Xuan Guo, Zhidong Zhao, Dongxiao He, and Di Jin. A survey on temporal interaction graph representation learning: Progress, challenges, and opportunities. *arXiv preprint arXiv:2505.04461*, 2025.

[3] Yejin Kim, Youngbin Lee, Minyoung Choe, Sungju Oh, and Yongjae Lee. Temporal graph networks for graph anomaly detection in financial networks. *arXiv preprint arXiv:2404.00060*, 2024.

[4] Haonan Yuan, Qingyun Sun, Xingcheng Fu, Ziwei Zhang, Cheng Ji, Hao Peng, and Jianxin Li. Environment-aware dynamic graph learning for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 36:49715–49747, 2023.

[5] Zeyang Zhang, Xin Wang, Ziwei Zhang, Zhou Qin, Weigao Wen, Hui Xue, Haoyang Li, and Wenwu Zhu. Spectral invariant learning for dynamic graphs under distribution shifts. *Advances in Neural Information Processing Systems*, 36:6619–6633, 2023.

[6] Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. Out-of-distribution generalization on graphs: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[7] Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. Learning invariant graph representations for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35:11828–11841, 2022.

[8] M Venkata Subbarao and P Samundiswary. Time-frequency analysis of non-stationary signals using frequency slice wavelet transform. In *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, pages 1–6. IEEE, 2016.

[9] Rakshit Trivedi, Mehrdad Farajtabar, Prasenjeet Biswal, and Hongyuan Zha. Dyrep: Learning representations over dynamic graphs. In *International conference on learning representations*, 2019.

[10] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[11] Brian Russell and Jiajun Han. Jean morlet and the continuous wavelet transform. *CREWES Res. Rep*, 28:115, 2016.

[12] Supriya Pandhre, Himangi Mittal, Manish Gupta, and Vineeth N Balasubramanian. Stwalk: learning trajectory representations in temporal graphs. In *Proceedings of the ACM India joint international conference on data science and management of data*, pages 210–219, 2018.

[13] Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1269–1278, 2019.

[14] Martin Strohmeier, Xavier Olive, Jannis Lübbe, Matthias Schäfer, and Vincent Lenders. Crowdsourced air traffic data from the opensky network 2019–2020. *Earth System Science Data*, 13(2):357–366, 2021.

[15] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998, 2008.

[16] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, pages 243–246, 2015.

[17] Kuansan Wang, Zhihong Shen, Chiyuan Huang, Chieh-Han Wu, Yuxiao Dong, and Anshul Kanakia. Microsoft academic graph: When experts are not enough. *Quantitative Science Studies*, 1(1):396–413, 2020.

[18] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[19] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1285–1293, 2012.

[20] Le Yu, Leilei Sun, Bowen Du, and Weifeng Lv. Towards better dynamic graph learning: New architecture and unified library. *Advances in Neural Information Processing Systems*, 36:67686–67700, 2023.

[21] Weilin Cong, Si Zhang, Jian Kang, Baichuan Yuan, Hao Wu, Xin Zhou, Hanghang Tong, and Mehrdad Mahdavi. Do we really need complicated model architectures for temporal networks? *arXiv preprint arXiv:2302.11636*, 2023.

[22] Anson Bastos, Abhishek Nadgeri, Kuldeep Singh, Toyotaro Suzumura, and Manish Singh. Learnable spectral wavelets on dynamic graphs to capture global interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6779–6787, 2023.

[23] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637*, 2020.

[24] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[25] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International conference on machine learning*, pages 5815–5826. PMLR, 2021.

[26] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

[27] Binwu Wang, Jiaming Ma, Pengkun Wang, Xu Wang, Yudong Zhang, Zhengyang Zhou, and Yang Wang. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2948–2959, 2024.

[28] Hyeokho Choi, Justin Romberg, Richard Baraniuk, and Nick Kingsbury. Hidden markov tree modeling of complex wavelet transforms. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)*, volume 1, pages 133–136. IEEE, 2000.

[29] Jonathan M Lilly and Sofia C Olhede. On the analytic wavelet transform. *IEEE transactions on information theory*, 56(8):4135–4156, 2010.

[30] Ingrid Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE transactions on information theory*, 36(5):961–1005, 2002.

[31] Jonathan M Lilly and Sofia C Olhede. Generalized morse wavelets as a superfamily of analytic wavelets. *IEEE Transactions on Signal Processing*, 60(11):6036–6041, 2012.

[32] HANG Su, QUAN Liu, and JINGSONG Li. Boundary effects reduction in wavelet transform for time-frequency analysis. *Wseas transactions on signal processing*, 8(4):169–179, 2012.

[33] Xinxun Zhang, Pengfei Jiao, Mengzhou Gao, Tianpeng Li, and Xuan Guo. Towards ood generalization in dynamic graphs via causal invariant learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 16379–16387, 2026.

[34] Qingyun Sun, Jiayi Luo, Haonan Yuan, Xingcheng Fu, Hao Peng, Jianxin Li, and Philip S Yu. Evolving graph learning for out-of-distribution generalization in non-stationary environments. *IEEE Transactions on pattern analysis and machine intelligence*, 2025.

[35] Guibin Zhang, Yiyan Qi, Ziyang Cheng, Yanwei Yue, Dawei Cheng, and Jian Guo. Rationalizing and augmenting dynamic graph neural networks. In *International Conference on Learning Representations*, volume 2025, pages 80317–80337, 2025.

[36] Tianjun Yao, Yongqiang Chen, Zhenhao Chen, Kai Hu, Zhiqiang Shen, and Kun Zhang. Empowering graph invariance learning with deep spurious infomax. *arXiv preprint arXiv:2407.11083*, 2024.

[37] Yuxing Tian, Yiyan Qi, and Fan Guo. Freedyg: Frequency enhanced continuous-time dynamic graph model for link prediction. In *International Conference on Learning Representations*, volume 2024, pages 8431–8450, 2024.

[38] Songgaojun Deng, Huzefa Rangwala, and Yue Ning. Dynamic knowledge graph based multi-event forecasting. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1585–1595, 2020.

[39] Jozef Barunik, Michael Ellington, et al. Dynamic networks in large financial and economic systems. *arXiv preprint arXiv:2007.07842*, 2020.

[40] Ziwei Zhang, Peng Cui, and Wenwu Zhu. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):249–270, 2020.

[41] Joakim Skarding, Bogdan Gabrys, and Katarzyna Musial. Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey. *iEEE Access*, 9:79143–79168, 2021.

[42] Yuecai Zhu, Fuyuan Lyu, Chengming Hu, Xi Chen, and Xue Liu. Encoder-decoder architecture for supervised dynamic graph learning: A survey. *arXiv preprint arXiv:2203.10480*, 2022.

[43] Chao Chen, Haoyu Geng, Nianzu Yang, Xiaokang Yang, and Junchi Yan. Easydgl: Encode, train and interpret for continuous-time dynamic graph learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10845–10862, 2024.

[44] Aravind Sankar, Yanhong Wu, Liang Gou, Wei Zhang, and Hao Yang. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks. In *Proceedings of the 13th international conference on web search and data mining*, pages 519–527, 2020.

[45] Yanbang Wang, Yen-Yu Chang, Yunyu Liu, Jure Leskovec, and Pan Li. Inductive representation learning in temporal networks via causal anonymous walks. *arXiv preprint arXiv:2101.05974*, 2021.

[46] Weining Shi, Zhisen Wen, Qinggang Zhang, Chentao Zhang, and Zhihong Zhang. Temporal graph thumbnail: Robust representation learning with global evolutionary skeleton. In *International Conference on Learning Representations*, volume 2026, pages 137964–137999, 2026.

[47] Defu Cao, Yujing Wang, Juanyong Duan, Ce Zhang, Xia Zhu, Congrui Huang, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, et al. Spectral temporal graph neural network for multivariate time-series forecasting. *Advances in neural information processing systems*, 33:17766–17778, 2020.

[48] Bingxin Zhou, Xinliang Liu, Yuehua Liu, Yunying Huang, Pietro Lio, and Yu Guang Wang. Well-conditioned spectral transforms for dynamic graph representation. In *Learning on Graphs Conference*, pages 12–1. PMLR, 2022.

[49] Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

[50] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4396–4415, 2022.

[51] Mucong Ding, Kezhi Kong, Jiuhai Chen, John Kirchenbauer, Micah Goldblum, David Wipf, Furong Huang, and Tom Goldstein. A closer look at distribution shifts and out-of-distribution generalization on graphs. 2021.

[52] Siqi Miao, Mia Liu, and Pan Li. Interpretable and generalizable graph learning via stochastic attention mechanism. In *International conference on machine learning*, pages 15524–15543. PMLR, 2022.

[53] Ling Yang, Jiayi Zheng, Heyuan Wang, Zhongyi Liu, Zhilin Huang, Shenda Hong, Wentao Zhang, and Bin Cui. Individual and structural graph information bottlenecks for out-of-distribution generalization. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):682–693, 2023.

[54] Yun Zhu, Haizhou Shi, Zhenshuo Zhang, and Siliang Tang. Mario: Model agnostic recipe for improving ood generalization of graph contrastive learning. In *Proceedings of the ACM Web Conference 2024*, pages 300–311, 2024.

[55] Xin Sun, Liang Wang, Qiang Liu, Shu Wu, Zilei Wang, and Liang Wang. Dive: subgraph disagreement for graph out-of-distribution generalization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2794–2805, 2024.

[56] H Xue, K Zhou, T Chen, K Guo, X Hu, Y Chang, and X Wang. Cap: co-adversarial perturbation on weights and features for improving generalization of graph neural networks. arxiv. 2021. *arXiv preprint arXiv:2110.14855*.

[57] Fuli Feng, Xiangnan He, Jie Tang, and Tat-Seng Chua. Graph adversarial training: Dynamically regularizing based on graph structure. *IEEE Transactions on Knowledge and Data Engineering*, 33(6):2493–2504, 2019.

[58] Alireza Sadeghi, Meng Ma, Bingcong Li, and Georgios B Giannakis. Distributionally robust semi-supervised learning over graphs. *arXiv preprint arXiv:2110.10582*, 2021.

[59] Yihan Wu, Aleksandar Bojchevski, and Heng Huang. Adversarial weight perturbation improves generalization in graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 10417–10425, 2023.

[60] Zhengyu Chen, Teng Xiao, Kun Kuang, Zheqi Lv, Min Zhang, Jinluan Yang, Chengqiang Lu, Hongxia Yang, and Fei Wu. Learning to reweight for generalizable graph neural network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 8320–8328, 2024.

[61] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*, 2019.

[62] Gilad Yehudai, Ethan Fetaya, Eli Meirom, Gal Chechik, and Haggai Maron. From local structures to size generalization in graph neural networks. In *International Conference on Machine Learning*, pages 11975–11986. PMLR, 2021.

[63] Mingchen Sun, Kaixiong Zhou, Xin He, Ying Wang, and Xin Wang. Gppt: Graph pre-training and prompt tuning to generalize graph neural networks. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1717–1727, 2022.

[64] Taoran Fang, Yunchao Zhang, Yang Yang, Chunping Wang, and Lei Chen. Universal prompt tuning for graph neural networks. *Advances in Neural Information Processing Systems*, 36:52464–52489, 2023.

[65] Haoyang Li, Xin Wang, Zeyang Zhang, Haibo Chen, Ziwei Zhang, and Wenwu Zhu. Disentangled graph self-supervised learning for out-of-distribution generalization. In *Forty-first International Conference on Machine Learning*, 2024.

[66] Yun Zhu, Yaoke Wang, Haizhou Shi, Zhenshuo Zhang, Dian Jiao, and Siliang Tang. Graphcontrol: Adding conditional control to universal graph pre-trained models for graph domain transfer learning. In *Proceedings of the ACM Web Conference 2024*, pages 539–550, 2024.

[67] Kuo Yang, Zhengyang Zhou, Qihe Huang, Limin Li, Yuxuan Liang, and Yang Wang. Improving generalization of dynamic graph learning via environment prompt. *Advances in Neural Information Processing Systems*, 37:70048–70075, 2024.

[68] Qi Zhu, Natalia Ponomareva, Jiawei Han, and Bryan Perozzi. Shift-robust gnns: Overcoming the limitations of localized graph training data. *Advances in Neural Information Processing Systems*, 34:27965–27977, 2021.

[69] Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. Ood-gnn: Out-of-distribution generalized graph neural network. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):7328–7340, 2022.

[70] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466*, 2022.

[71] Yuntao Du, Jindong Wang, Wenjie Feng, Sinno Pan, Tao Qin, Renjun Xu, and Chongjun Wang. Adarnn: Adaptive learning and forecasting of time series. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 402–411, 2021.

[72] Jean-Christophe Gagnon-Audet, Kartik Ahuja, Mohammad-Javad Darvishi-Bayazi, Pooneh Mousavi, Guillaume Dumas, and Irina Rish. Woods: Benchmarks for out-of-distribution generalization in time series. *arXiv preprint arXiv:2203.09978*, 2022.

[73] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. Invariant rationalization. In *International Conference on Machine Learning*, pages 1448–1458. PMLR, 2020.

[74] Beatrice Bevilacqua, Yangze Zhou, and Bruno Ribeiro. Size-invariant graph representations for graph classification extrapolations. In *International Conference on Machine Learning*, pages 837–851. PMLR, 2021.

[75] Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. Learning causally invariant representations for out-of-distribution generalization on graphs. *Advances in Neural Information Processing Systems*, 35:22131–22148, 2022.

[76] Yongduo Sui, Xiang Wang, Jiancan Wu, Min Lin, Xiangnan He, and Tat-Seng Chua. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1696–1705, 2022.

[77] Zhengyang Zhou, Qihe Huang, Kuo Yang, Kun Wang, Xu Wang, Yudong Zhang, Yuxuan Liang, and Yang Wang. Maintaining the status quo: Capturing invariant relations for ood spatiotemporal learning. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 3603–3614, 2023.

[78] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

[79] Praveen Venkateswaran, Vinod Muthusamy, Vatche Isahagian, and Nalini Venkatasubramanian. Environment agnostic invariant risk minimization for classification of sequential datasets. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 1615–1624, 2021.

[80] Shaohua Fan, Xiao Wang, Yanhu Mo, Chuan Shi, and Jian Tang. Debiasing graph neural networks via learning disentangled causal substructure. *Advances in Neural Information Processing Systems*, 35:24934–24946, 2022.

[81] Yongqiang Chen, Yatao Bian, Kaiwen Zhou, Binghui Xie, Bo Han, and James Cheng. Does invariant graph learning via environment augmentation learn invariance? *Advances in Neural Information Processing Systems*, 36:71486–71519, 2023.

[82] Shurui Gui, Meng Liu, Xiner Li, Youzhi Luo, and Shuiwang Ji. Joint learning of label and environment causal independence for graph out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 36:3945–3978, 2023.

[83] Shaohua Fan, Xiao Wang, Chuan Shi, Peng Cui, and Bai Wang. Generalizing graph neural networks on out-of-distribution graphs. *IEEE transactions on pattern analysis and machine intelligence*, 46(1):322–337, 2023.

[84] Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. Discovering invariant rationales for graph neural networks. *arXiv preprint arXiv:2201.12872*, 2022.

[85] Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. Invariance principle meets out-of-distribution generalization on graphs. In *ICML 2022: workshop on spurious correlations, invariance and stability*, 2022.

[86] Jianxin Ma, Peng Cui, Kun Kuang, Xin Wang, and Wenwu Zhu. Disentangled graph convolutional networks. In *International conference on machine learning*, pages 4212–4221. PMLR, 2019.

[87] Yanbei Liu, Xiao Wang, Shu Wu, and Zhitao Xiao. Independence promoted graph disentangled networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 4916–4923, 2020.

[88] Yiding Yang, Zunlei Feng, Mingli Song, and Xinchao Wang. Factorizable graph convolutional networks. *Advances in Neural Information Processing Systems*, 33:20286–20296, 2020.

[89] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, and Wenwu Zhu. Out-of-distribution generalized dynamic graph neural network with disentangled intervention and invariance promotion. *arXiv preprint arXiv:2311.14255*, 2023.

[90] Xiaojie Guo, Liang Zhao, Zhao Qin, Lingfei Wu, Amarda Shehu, and Yanfang Ye. Interpretable deep graph generation with node-edge co-disentanglement. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1697–1707, 2020.

[91] Haoyang Li, Xin Wang, Ziwei Zhang, Zehuan Yuan, Hang Li, and Wenwu Zhu. Disentangled contrastive learning on graphs. *Advances in Neural Information Processing Systems*, 34:21872–21884, 2021.

[92] Haoyang Li, Ziwei Zhang, Xin Wang, and Wenwu Zhu. Disentangled graph contrastive learning with independence promotion. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7856–7869, 2022.

[93] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

[94] Kun Yi, Qi Zhang, Wei Fan, Longbing Cao, Shoujin Wang, Hui He, Guodong Long, Liang Hu, Qingsong Wen, and Hui Xiong. A survey on deep learning based time series analysis with frequency transformation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6206–6215, 2025.

[95] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pages 27268–27286. PMLR, 2022.

[96] Alan V Oppenheim and Jae S Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 2005.

[97] Bruce C Hansen and Robert F Hess. Structural sparseness and spatial phase alignment in natural scenes. *Journal of the Optical Society of America A*, 24(7):1873–1885, 2007.

[98] Xiang Zhang, Ziyuan Zhao, Theodoros Tsiligkaridis, and Marinka Zitnik. Self-supervised contrastive pre-training for time series via time-frequency consistency. *Advances in neural information processing systems*, 35:3988–4003, 2022.

[99] Gaurav Gupta, Xiongye Xiao, and Paul Bogdan. Multiwavelet-based operator learning for differential equations 2021. *arXiv preprint arXiv:2109.13459*.

[100] Bingbing Xu, Huawei Shen, Qi Cao, Yunqi Qiu, and Xueqi Cheng. Graph wavelet neural network. *arXiv preprint arXiv:1904.07785*, 2019.

[101] Henry Kenlay, Dorina Thanou, and Xiaowen Dong. Interpretable stability bounds for spectral graph filters. In *International conference on machine learning*, pages 5388–5397. PMLR, 2021.

[102] Christopher Torrence and Gilbert P Compo. A practical guide to wavelet analysis. *Bulletin of the American Meteorological society*, 79(1):61–78, 1998.

Contents of Appendix

A Frequently Asked Questions (FAQs) 18

B Method Details 23

B.1 Fundamentals of Discrete Complex Morlet Wavelet Theory 23

B.1.1 Time-Frequency Atoms and Heisenberg Boxes 23

B.1.2 Construction of Continuous Analytic Wavelets 24

B.1.3 From Continuous to Discrete 24

B.2 Discrete Complex Wavelet Transform and Feature Mapping 25

B.3 The Wavelet Coefficient-Domain Invariant Learning Module 26

C Additional Experiments 29

C.1 Quantitative Results of Ablation Studies 29

C.2 Hyperparameter Sensitivity Analysis 30

D Implementation Details 34

D.1 Task Definitions 34

D.2 Evaluation Metrics 34

D.3 Datasets Construction and OOD Splitting Details 35

D.4 Baseline Details 36

E Case Study 38

E.1 Limitations of Spectral-Domain Baseline Methods 38

E.2 Fine-Grained Disentanglement of WavCIL in the Wavelet Coefficient Domain 39

F Limitations 39

G Related Works 39

H Broader Impacts 40

Algorithm 1 Training pipeline for WavCIL

Require: Dynamic graph sequence $\mathcal{G}_{1:T}$, Target labels Y, Interventional times I.
Require: Hyperparameters: Scales K, central frequency ω_0, temperature τ, penalty λ, variance weight β.
Ensure: The causal node representations $\mathbf{z}_c \in \mathbb{R}^{N \times d}$ for downstream tasks.

for epoch $= 1, 2, \ldots,$ MaxEpoch **do**
  Stage 1: Spatio-Temporal Node Trajectory Encoding
  $\mathbf{Z} \leftarrow \text{Encoder}(\mathcal{G}_{1:T})$ ▷ Encode spatio-temporal trajectories
  Stage 2: Discrete Complex Wavelet Transform (DCWT)
  $\mathbf{Z}_{fft} \leftarrow \text{FFT}(\mathbf{Z})$ ▷ 1D FFT along the temporal axis
  for $k = 1$ **to** K **do**
    $\psi_{fft,s_k} \leftarrow \text{FFT}(\psi_{s_k})$ ▷ Complex Morlet basis via FFT
    $\mathcal{W}_{v,k} \leftarrow \text{IFFT}(\mathbf{Z}_{fft} \odot \psi_{fft,s_k})$ ▷ Fast circular convolution
  end for
  Stage 3: Adaptive Mask Generation & Feature Disentanglement
  $\mathbf{H}_{in} \leftarrow \text{Concat}(\text{Re}(\mathcal{W}_v), \text{Im}(\mathcal{W}_v))$ ▷ Stack real & imaginary components
  $\mathbf{L} \leftarrow \Phi_{mask}(\mathbf{H}_{in})$ ▷ Compute latent importance logits
  $\mathbf{M}_c \leftarrow \sigma(\mathbf{L}/\tau), \quad \mathbf{M}_s \leftarrow 1 - \mathbf{M}_c$ ▷ Temperature-scaled dual masks
  $\mathcal{W}_c \leftarrow \mathcal{W}_v \odot \mathbf{M}_c, \quad \mathcal{W}_s \leftarrow \mathcal{W}_v \odot \mathbf{M}_s$ ▷ Coefficient-Domain filtering
  $\mathbf{z}_c \leftarrow \Phi_{proj}(\text{Flatten}(\mathcal{W}_c))$ ▷ Extract stable causal representation
  $\mathbf{z}_s \leftarrow \Phi_{proj}(\text{Flatten}(\mathcal{W}_s))$ ▷ Extract spurious representation
  Stage 4: Invariance Optimization & Environmental Intervention
  $\mathcal{L}_{task} \leftarrow \ell_{\text{CE}}(f_{pred}(\mathbf{z}_c), Y)$ ▷ Empirical Risk Minimization (ERM)
  $\mathcal{E}_{loss} \leftarrow \emptyset$ ▷ Init interventional loss collector
  for $i = 1$ **to** I **do**
    $\tilde{\mathbf{z}}_s^{(i)} \leftarrow \text{Permute}(\mathbf{z}_s)$ ▷ Randomly sample spurious environments
    $\mathcal{E}_{loss} \leftarrow \mathcal{E}_{loss} \cup \{\ell_{\text{CE}}(f_{pred}(\mathbf{z}_c, \tilde{\mathbf{z}}_s^{(i)}), Y)\}$ ▷ Interventional prediction loss
  end for
  $\mathcal{L}_{env} \leftarrow \text{Mean}(\mathcal{E}_{loss}) + \beta \cdot \text{Var}(\mathcal{E}_{loss})$ ▷ Invariant variance penalty
  $\mathcal{L}_{total} \leftarrow \mathcal{L}_{task} + \lambda \cdot \mathcal{L}_{env}$ ▷ Hybrid invariant objective
  Update parameters via gradient descent on $\mathcal{L}_{total}$ ▷ Network parameters optimization
end for

A Frequently Asked Questions (FAQs)

A.1 Code and Data Availability

To ensure the transparency and reproducibility of our research, the code of WavCIL is publicly available at https://github.com/yantongzhu/WavCIL. This repository and the supplementary materials include the complete implementation of WavCIL, along with comprehensive scripts for model training and evaluation. Additionally, all dynamic graph datasets utilized in our experiments are provided or properly linked within the repository, ensuring that researchers have full access to the resources required to reproduce and extend our work.

A.2 Computational Complexity Analysis

In this section, we perform a rigorous computational complexity analysis in Algorithm 1 and provide empirical validation of overall model efficiency. The full forward propagation and intervention optimization pipeline of our proposed WavCIL learning framework is summarized in Algorithm 1. To facilitate theoretical derivation, we define the input dynamic graph with N nodes and $|\mathcal{E}|$ edges, a temporal sequence length of T, node feature dimension d, hidden dimension d', number of discrete wavelet scales K, central frequency ω_0, temperature τ, penalty λ, variance weight β, number of spatial backbone encoder layers L, and number of environment interventions I. The time complexity of a single training iteration of WavCIL can be rigorously decomposed into distinct terms as follows:

- **Line 2–3 (Node Trajectory Encoding):** An L-layer graph neural network (e.g., GCN or GAT) is applied across the entire sequence of T temporal snapshots to obtain the resulting temporal trajectory $\mathbf{Z} \in \mathbb{R}^{N \times T \times d}$. The overall complexity of this specific stage depends on the sparsity of the graph structure, with a total estimated computational cost of $\mathcal{O}\big(L \cdot T \cdot (Ndd' + |\mathcal{E}|d)\big)$.

- **Line 4–9 (Discrete Complex Wavelet Transform):** This module is the fundamental core of temporal dimension acceleration in WavCIL. Performing the FFT on the trajectory $\mathbf{Z}$ along the temporal dimension yields a complexity of $\mathcal{O}(N \cdot d \cdot T \log T)$. In the frequency domain, wavelet bases are generated for each of the K different scales, followed by element-wise multiplicative convolution, and the IFFT is used to transform the features back to the time domain. The total time complexity of this loop is strictly controlled with $\mathcal{O}\big(K \cdot N \cdot d \cdot T \log T\big)$.
- **Line 10–16 (Mask Generation and Feature Disentanglement):** After extracting the specific observed tail window of the complex wavelet coefficients and concatenating their respective amplitude-phase (real-imaginary) components, a corresponding dual mask is generated via the dedicated MLP, with a complexity of $\mathcal{O}(N \cdot K \cdot d \cdot d')$. Subsequently, the process of applying mask filtering and mapping back to $N \times d$ real-valued causal and spurious representations via the linear projection layer Φ_{proj}, with the same projection cost of $\mathcal{O}(N \cdot K \cdot d \cdot d')$.
- **Line 17–27 (Invariance Optimization and Intervention):** We conduct I random permutations on the specific extracted spurious features and feed them into the underlying causal predictor f_{pred} in a forward pass to compute the resulting variance penalty. As the predictor is usually a simple linear mapping, the computational complexity of this step is $\mathcal{O}(I \cdot N \cdot d)$.

Combining the above procedures, the overall time complexity of an L-layer WavCIL architecture is:

$$\mathcal{O}\Big(L \cdot T \cdot (Ndd' + |\mathcal{E}|d) + K \cdot N \cdot d \cdot T \log T + N \cdot K \cdot d \cdot d' + I \cdot N \cdot d\Big). \tag{9}$$

Although WavCIL introduces extra wavelet mapping and dual-mask generation, both the number of scales K and the number of interventions I are small constants. Consequently, its overall core time complexity with respect to the temporal length T is specifically dominated by the efficient Fast Fourier Transform (FFT), i.e., $\mathcal{O}(T \log T)$. This successfully breaks the potential computational explosion bottleneck of $\mathcal{O}(T^2)$ complexity suffered by traditional dynamic graph models with standard spatio-temporal attention (e.g., DyGFormer [20]) when processing long sequences.

Nevertheless, real computational efficiency remains a critical bottleneck for the practical deployment of various dynamic graph models. Theoretical complexity lower bounds alone are often not sufficiently persuasive. For further empirical validation, we record the average training time per epoch of the proposed model across all standard benchmark evaluation datasets.

Table 3: Training time per epoch (seconds) comparison across datasets. WavCIL achieves efficiency comparable to the spectral baseline SILD while being orders of magnitude faster than standard spatiotemporal models like DyGFormer.

Training Time per Epoch (Seconds)							
	Real-world Link Prediction				Synthetic Node Classification		
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
DyGFormer	40.398±0.905	232.814±26.292	286.227±12.316	544.104±137.356	2564.312±43.378	2521.409±85.477	2451.583±53.870
SILD	1.082±0.088	2.181±0.152	2.821±0.258	2.886±0.281	1.321±0.166	1.273±0.154	1.266±0.074
DyCIL	0.505±0.003	0.534±0.028	0.538±0.048	0.534±0.005	2.692±0.019	2.653±0.120	2.606±0.154
DIDA	1.072±0.046	1.212±0.161	1.269±0.152	1.208±0.138	OOM	OOM	OOM
EAGLE	5.382±2.072	6.212±0.536	3.490±0.073	3.490±0.073	0.941±0.113	0.830±0.034	0.816±0.027
WavCIL	1.372±0.088	2.831±0.424	3.132±0.202	2.753±0.157	1.289±0.050	1.363±0.217	1.201±0.151

Training efficiency analysis: As shown in Table 3, compared with DyGFormer [20], which has strong spatio-temporal representation power but is limited by $\mathcal{O}(T^2)$ attention computation, WavCIL achieves an order-of-magnitude improvement in computational efficiency (e.g., on Reddit, DyGFormer takes 286.227s while WavCIL only requires 3.132s). More critically, even with the introduction of multi-scale complex-plane feature computations, the overall training time of WavCIL remains the same level with that of the current efficient spectral-domain baseline SILD [5] (e.g., on Wiki [13], SILD takes 1.082s, while WavCIL takes 1.372s). This empirically demonstrates that WavCIL successfully incorporates a complex local time-frequency disentanglement mechanism without sacrificing training scalability, making it hold full potential for deployment in real-world large-scale dynamic graphs.

Inference efficiency analysis: Regarding inference efficiency, as shown in Table 4, the current inference time of WavCIL is higher than that of lightweight methods such as EAGLE. This overhead primarily stems from computing localized multi-scale DCWT coefficients and adaptive masks across multiple forecast origins. We note that the inference stage does not involve the training-only

Table 4: Inference time per snapshot (milliseconds) comparison.

Inference Time per Snapshot (ms)							
	Real-world Link Prediction				Synthetic Node Classification		
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
DyGFormer	238.697±1.731	154.754±3.627	173.357±44.630	649.196±166.095	60.826±1.911	60.748±2.610	57.912±0.400
SILD	416.740±33.116	96.262±23.308	536.416±144.350	486.635±59.277	512.959±271.443	399.775±294.422	439.726±108.358
DyCIL	139.637±9.984	142.608±1.248	149.354±4.884	146.452±2.892	1211.755±62.656	1201.734±55.783	1185.366±86.348
DIDA	243.388±58.326	367.848±77.835	394.641±109.078	394.364±91.230	OOM	OOM	OOM
EAGLE	25.325±8.452	45.801±6.563	30.852±0.817	61.421±1.814	15.687±6.366	11.976±0.430	12.017±0.816
WavCIL	817.320±15.960	797.012±35.416	1041.837±111.335	852.274±90.885	584.802±194.174	453.197±121.493	517.591±270.990

intervention operations, and the latency can potentially be reduced in the future through caching wavelet bases or pruning the mask network.

Table 5: Peak GPU memory usage (GB) during training. OOM indicates out-of-memory on a 24GB GPU.

Peak GPU Memory (GB)							
	Real-world Link Prediction				Synthetic Node Classification		
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
DyGFormer	1.002±0.001	1.122±0.001	1.226±0.001	1.296±0.013	7.395±0.000	7.393±0.000	7.394±0.000
SILD	1.047±0.000	1.666±0.000	2.788±0.000	2.195±0.000	4.521±0.000	4.521±0.000	4.522±0.000
DyCIL	0.534±0.001	0.795±0.001	0.777±0.000	0.585±0.000	18.510±0.002	18.498±0.003	18.498±0.006
DIDA	1.122±0.000	4.886±0.000	6.196±0.000	5.408±0.000	OOM	OOM	OOM
EAGLE	3.064±0.000	6.467±0.000	4.572±0.000	8.275±0.000	1.944±0.000	1.944±0.000	1.944±0.000
WavCIL	1.257±0.000	2.053±0.000	2.863±0.000	2.236±0.000	10.147±0.000	7.225±0.000	7.225±0.000

Memory footprint analysis: In terms of memory consumption, as shown in Table 5, WavCIL maintains a peak GPU memory usage comparable to SILD across most datasets. On the synthetic node classification tasks, WavCIL exhibits substantially lower memory footprint than DyCIL; for example, at the shift level of 0.8, DyCIL consumes 18.498GB while WavCIL uses only 7.225GB.. Compared with DIDA, WavCIL consumes less memory on Flight, Reddit, and MOOC. These results demonstrate that WavCIL achieves superior OOD generalization while maintaining a moderate GPU memory footprint, making it suitable for GPU-constrained deployment scenarios.

Training time consistency analysis: In addition to average runtime, we evaluate training-time consistency via the coefficient of variation (CV) of per-epoch training time, defined as:

$$\mathrm{CV} = \frac{\mathrm{Std}(T_{\mathrm{epoch}})}{\mathrm{Mean}(T_{\mathrm{epoch}})}.$$

Lower CV indicates more stable per-epoch runtime. As shown in Table 6, WavCIL achieves the lowest or second-lowest CV in most settings, demonstrating predictable training efficiency.

Table 6: Coefficient of variation (CV, %) of training time per epoch across datasets.

CV of Training Time per Epoch (%)							
	Real-world Link Prediction				Synthetic Node Classification		
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
SILD	45.980±8.573	25.938±5.767	14.687±4.713	25.938±5.767	36.449±10.898	45.243±13.619	42.622±9.133
DyCIL	80.768±3.975	76.410±10.469	80.738±3.970	82.833±4.030	36.597±8.055	44.681±1.511	39.437±9.257
DIDA	38.828±17.504	31.030±12.912	30.223±12.018	34.810±10.623	OOM	OOM	OOM
EAGLE	14.629±1.764	10.506±0.814	10.187±0.563	16.150±0.389	57.544±0.682	57.700±0.375	56.759±0.515
WavCIL	21.318±0.974	13.160±4.980	10.221±4.849	14.642±1.145	24.959±10.956	43.697±3.177	37.798±8.217

A.3 What assumptions justify interpreting the learned wavelet coefficient masks as causal and spurious components?

We provide further clarification here on the causal-invariant formulation, the role of our objective, the scope of identifiability, and the empirical evidence supporting our interpretation.

Assumptions underlying the causal–spurious interpretation. Following prior dynamic graph OOD works such as DIDA [1], SILD [5], and EAGLE [4], we adopt a task-level causal-invariant formulation. DIDA and SILD decompose a dynamic-graph pattern into invariant P_I and variant P_V components:

$$Y = f(P_I) + \epsilon, \qquad Y \perp P_V \mid P_I. \tag{10}$$

EAGLE expresses the same invariant-and-sufficient principle as:

$$p(Y \mid P_I, e) = p(Y \mid P_I), \qquad Y \perp P_V \mid P_I, \tag{11}$$

where e denotes the environment. Under this formulation, EAGLE interprets P_I and P_V as causal and spurious components, respectively [4]. Specifically, causal/invariant information is label-sufficient and predictively stable across environments, whereas spurious/variant information may predict well within one environment but varies across environments. WavCIL realizes this decomposition in the wavelet coefficient domain via complementary masks:

$$\mathcal{W}_c = \mathbf{M}_c \odot \mathcal{W}, \qquad \mathcal{W}_s = (1 - \mathbf{M}_c) \odot \mathcal{W}. \tag{12}$$

This interpretation rests on four assumptions: (1) the invariant component P_I sufficiently predicts the label Y; (2) the variant component P_V adds no additional label-relevant information given P_I; (3) the relationship between P_I and Y remains stable across environments; (4) invariant and variant patterns are separable in the localized time-frequency representations provided by the wavelet transform.

Role of the optimization objective. In WavCIL, the semantics of the two branches are jointly encouraged by three complementary components.

First, the complementary masks exhaustively allocate coefficients via $\mathbf{M}_c + \mathbf{M}_s = \mathbf{1}$, with temperature scaling encouraging near-binary allocation. Second, the task loss $\mathcal{L}_{task} = \mathbb{E}[\ell(f_{pred}(\mathbf{Z}_c), Y)]$ discourages an empty or label-insufficient causal branch. Third, the intervention-based variance penalty (Equation 36 in the main text) keeps $\mathbf{Z}_c$ fixed, intervenes on the complementary spurious representation, and penalizes the variance of intervention risks. Any environment-sensitive content in $\mathbf{Z}_c$ would increase this variance. The objective therefore encourages stability under environmental changes, consistent with the invariance principles in DIDA [1] and SILD [5]. Thus, the causal interpretation is supported jointly by predictive sufficiency and intervention stability, rather than by mask values alone.

Scope of identifiability. WavCIL adopts **task-functional identifiability**: masks that assign redundant coefficients differently but yield the same label-sufficient, intervention-invariant representation and OOD predictions are considered task-equivalent. Under the assumption of non-overlapping invariant and variant spectral supports, SILD [5] proves the existence of a mask with bounded prediction error as variant patterns change. WavCIL extends this principle from global Fourier analysis to localized wavelet representations, motivating robust partitioning under separability.

Empirical support from ablation studies. As detailed in Appendix C.1, removing disentanglement masks (w/o DM) degrades 13/14 tasks. On Synthetic 0.4/0.6/0.8, it reduces link-prediction AUC by 5.21/5.33/5.93 points and node-classification accuracy by 12.09/8.32/11.96 points. Removing invariant learning (w/o IL) degrades 10/14 tasks. These results support the effectiveness and non-triviality of adaptive allocation and intervention regularization.

A.4 Why is it necessary to employ complex wavelets? Does concatenating the real and imaginary parts contradict the original intention of amplitude-phase separation?

This operation not only does not contradict the purpose of amplitude-phase separation, but also represents the core reason why complex wavelets are indispensable for modeling dynamic graph evolution. Complex wavelets serve as the fundamental mathematical tool that simultaneously satisfy the requirements of **time-frequency localization** and **representation stability**.

Temporal Localization Beyond Global Fourier Transform. While the conventional Fourier transform (e.g., as used in SILD [5]) can extract frequency patterns of evolution, its global integral property sacrifices the ability to localize specific events in time dimension [5]. For this reason, we adopt wavelet analysis to achieve joint time-frequency localization.

Overcoming translation variance. If one directly uses the computationally cheaper real wavelets (e.g., standard Haar or real Gabor), the resulting extracted features suffer from severe "shift variance": a small shift of the underlying temporal evolution pattern along the time axis leads to dramatic oscillations in the amplitudes of the real wavelet coefficients. Such unstable signal input prevents the downstream mask generator from learning consistent disentanglement rules [28].

Rigorous Amplitude-Phase Separation and Encoding. Complex analytic wavelets strictly separate the energy and structure of signals through their orthogonal real and imaginary components. In neural network frameworks, concatenating the real (Re) and imaginary (Im) parts along the feature dimension is a standard, lossless paradigm for processing complex-domain signals. This operation fully preserves the network's ability to compute instantaneous amplitude (representing the intensity of abrupt changes) and instantaneous phase (representing the location of evolution). This "magnitude-plus-phase" input endows the model with approximate translation invariance, forming the mathematical foundation that enables the masking network to stably identify causal and spurious patterns [29].

A.5 Does the performance improvement of WavCIL arise from its time-frequency disentanglement mechanism, or merely from increased "model capacity" introduced by the extra mask and projection layers?

The substantial performance leap of our proposed WavCIL fundamentally originates from the intrinsic superior fine-grained time-frequency localization capability comprehensively exhibited in the wavelet domain, rather than mere simple parameter expansion or engineering tricks [30].

Strict Capacity Alignment. To eliminate the interference of the model's parameter count, we introduce a Strictly-Aligned FFT baseline. This variant replaces the discrete complex wavelet transform (DCWT) with global fast Fourier transform (FFT), while retaining exactly the same adaptive mask architecture, projection layer dimensions, and optimization objectives as WavCIL. This setup ensures the two models possess strictly equivalent theoretical fitting capacity.

Table 7: A comprehensive performance comparison between our proposed WavCIL, the capacity-aligned FFT baseline, and the corresponding w/o DCWT variant on standard Link Prediction and Node Classification tasks (↓ / ↑ indicates performance drop/improvement compared to WavCIL).

Link Prediction

Variant	Wiki		Flight		Reddit		MOOC		Synthetic0.4		Synthetic0.6		Synthetic0.8	
	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ
WavCIL (Ours)	85.76±0.64	-	97.13±0.09	-	90.04±0.70	-	71.26±4.71	-	90.87±0.33	-	90.72±0.23	-	91.01±0.11	-
w/o DCWT	79.92±1.69	↓5.84	95.34±1.70	↓1.79	87.95±0.92	↓2.09	68.64±10.86	↓2.62	77.11±0.98	↓13.76	72.56±1.10	↓18.16	71.22±0.63	↓19.79
Strictly-Aligned FFT	72.31±2.03	↓13.45	96.59±0.09	↓0.54	85.69±2.07	↓4.35	69.53±1.19	↓1.73	84.32±1.50	↓6.55	80.02±0.74	↓10.70	78.51±2.14	↓12.50

Node Classification

Variant	Aminer15		Aminer16		Aminer17		OGBN-Arxiv		Synthetic0.4		Synthetic0.6		Synthetic0.8	
	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ
WavCIL (Ours)	51.88±0.76	-	53.78±0.41	-	45.57±1.01	-	58.53±0.68	-	43.89±0.36	-	42.32±4.03	-	41.55±3.69	-
w/o DCWT	51.60±0.73	↓0.28	55.07±0.35	↑1.29	45.17±0.79	↓0.40	55.10±1.29	↓3.43	41.98±3.67	↓1.91	40.48±2.18	↓1.84	37.91±1.36	↓3.64
Strictly-Aligned FFT	47.24±1.19	↓4.64	49.12±1.91	↓4.66	40.43±1.04	↓5.14	56.07±2.41	↓2.46	39.38±3.56	↓4.51	32.06±2.17	↓10.26	34.47±0.30	↓7.08

Superiority of Mathematical Tools. As illustrated in Table 7, WavCIL still substantially outperforms the standard frequency-domain baseline even under tightly aligned model capacity (e.g., under the extreme Synthetic 0.8 shift, WavCIL achieves an AUC advantage of over 12%). This strongly demonstrates that, because Fourier analysis loses temporal locality, blindly increasing the capacity of the masks cannot effectively separate deeply entangled transient shifts from the underlying time domain [30]. The superiority of WavCIL essentially arises from the intrinsic fundamental mathematical advantage provided by the DCWT. Full quantitative results and performance degradation analysis for ablation variants on the full dataset are provided in Appendix C.1.

A.6 What role does the disentangled mask $\mathbf{M}_c$ play? Is there evidence that it is more than just a black-box attention mechanism?

$\mathbf{M}_c$ is not an uninterpretable black-box attention mechanism, but rather represents an explicit, mathematically exact local filter with a clear physical topology in the joint time-frequency space.

Intuitive Visual Evidence. To provide straightforward and visually convincing evidence for the underlying mechanism, we present visualizations of the disentanglement process. The visualization case study is provided in Appendix E. The visualizations show how the model operates when facing bursts of local interactions on real-world MOOC [13] dynamic graph data.

The Case Study visualizations clearly show that traditional global spectral-domain baselines erroneously propagate transient noise to unperturbed time steps when encountering local distribution shifts, resulting in severe temporal leakage and contamination. In contrast, thanks to the joint time-frequency localization property of DCWT, the mask $\mathbf{M}_c$ in WavCIL can accurately locate underlying

spurious patterns directly on the wavelet coefficients. This physical isolation phenomenon at the topological level dispels concerns over the black-box nature of the parameterized mask.

B Method Details

This section provides a comprehensive and detailed supplement to the rigorous mathematical derivations and the associated specific algorithmic implementation details of the Wavelet Coefficient Invariant Learning (WavCIL) framework previously introduced in Section 3 of the main text.

B.1 Fundamentals of Discrete Complex Morlet Wavelet Theory

In Section 3.1 of the main text, we introduce the Discrete Complex Morlet Wavelet to simultaneously capture time-localized shifts and causal patterns in the non-stationary evolution of dynamic graphs. Starting from the fundamental theorems of time-frequency analysis, this subsection presents a detailed derivation of the mathematical motivations for constructing this wavelet [30].

B.1.1 Time-Frequency Atoms and Heisenberg Boxes: Achieving Optimal Time-Frequency Localization

The core of linear time-frequency transforms lies in computing the mathematical interaction between a specific underlying signal and a set of waveforms that are simultaneously highly concentrated in both the time and frequency domains (i.e., time-frequency atoms). Let $\psi \in L^2(\mathbb{R})$ be an elementary time-frequency atom satisfying the standard normalization condition $|\psi|^2 = \int_{-\infty}^{+\infty} |\psi(t)|^2 dt = 1$.

In time-frequency analysis, the localized slice of information extracted by ψ can be represented as a region in the joint time-frequency plane (t, ω), known as the Heisenberg Box. By regarding $|\psi(t)|^2$ as a probability distribution, its energy center u in the time domain is defined as:

$$u = \int_{-\infty}^{+\infty} t|\psi(t)|^2 dt. \tag{13}$$

The spread (i.e., variance) along the time axis, denoted σ_t^2, is measured as:

$$\sigma_t^2 = \int_{-\infty}^{+\infty} (t-u)^2 |\psi(t)|^2 dt. \tag{14}$$

Similarly, by Plancherel's theorem, in the frequency domain we have $\int_{-\infty}^{+\infty} |\hat{\psi}(\omega)|^2 d\omega = 2\pi$. The resultant frequency center ξ and associated frequency variance σ_ω^2 are respectively defined as:

$$\xi = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \omega |\hat{\psi}(\omega)|^2 d\omega, \tag{15}$$

$$\sigma_\omega^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} (\omega - \xi)^2 |\hat{\psi}(\omega)|^2 d\omega. \tag{16}$$

The rectangular region defined by the above parameters constitutes the Heisenberg Box. However, according to the Heisenberg Uncertainty Principle, time-frequency atoms cannot be infinitely precise in both time and frequency simultaneously. The area of this rectangle has a theoretical lower bound:

$$\sigma_t \sigma_\omega \geq \frac{1}{2}. \tag{17}$$

Among all possible candidate window functions, only the Gaussian function can uniquely make this time-frequency area reach the minimum theoretical lower bound of $1/2$. This fundamental mathematical theorem provides the most direct theoretical motivation for our choice of the Gaussian envelope $\exp(-t^2/2\sigma^2)$ in the WavCIL main text, ensuring that the model attains the theoretically optimal joint time-frequency resolution when processing dynamic graphs [10].

B.1.2 Construction of Continuous Analytic Wavelets: Fine-Grained Disentanglement of Causal and Spurious Patterns

With the optimal envelope established, to analyze the time-varying evolution of local frequencies in dynamic graphs, we must be able to separate the amplitude and phase information of the signal. This necessitates the employment of complex analytic wavelets [29, 31]. Mathematically, the Fourier transform of an analytic signal is strictly zero at negative frequencies:

$$\hat{\psi}(\omega) = 0, \quad \text{if } \omega < 0. \tag{18}$$

Analytic wavelets can be constructed by performing frequency modulation on a real-valued and symmetric window function $g(t)$ according to the following standard mathematical procedure:

$$\psi(t) = g(t)\exp(j\eta t). \tag{19}$$

The resulting Fourier transform is analytically given by $\hat{\psi}(\omega) = \hat{g}(\omega - \eta)$. Building directly on the previous theoretical conclusion, when $g(t)$ is specifically chosen as the Gaussian window, we obtain the classic Gabor wavelet (which forms the fundamental basis of the Morlet wavelet). As long as the condition $\sigma^2\eta^2 \gg 1$ holds — i.e., the numerical product of the bandwidth and the center frequency is sufficiently large — the energy of $\hat{g}(\omega)$ in the negative frequency region is approximately zero.

Building directly upon the aforementioned mother wavelet $\psi(t)$ as a primary foundation, we can now formally construct the complete family of basis functions for the continuous analytic wavelet by effectively introducing both the scale parameter $s \in \mathbb{R}^+$ and the translation parameter $u \in \mathbb{R}$:

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right) = \frac{1}{\sqrt{s}}g\left(\frac{t-u}{s}\right)\exp\left(j\eta\frac{t-u}{s}\right). \tag{20}$$

This continuous mathematical construction successfully guarantees the analyticity of the basis functions, thereby effectively endowing the overall WavCIL framework with the inherent theoretical capability to actually disentangle causal periodic patterns from the phase component while also consistently extracting the intensity of environmental abrupt changes from the amplitude component.

B.1.3 From Continuous to Discrete: Avoiding Redundancy and Computational Explosion

Although the Continuous Wavelet Transform (CWT) provides a smooth time-frequency representation, it nevertheless suffers from two critical drawbacks in deep learning frameworks for dynamic graphs, which directly motivate our final adoption of the Discrete Complex Wavelet Transform (DCWT):

- **High Representational Redundancy:** The CWT maps a one-dimensional time signal to a two-dimensional continuous time-frequency surface, which is often found to be inherently highly redundant. This redundancy is governed by the Reproducing Kernel Equation:

$$Wf(u_0, s_0) = \frac{1}{C_\psi}\int_{-\infty}^{+\infty} K(u, u_0, s, s_0)Wf(u, s)du\frac{ds}{s^2}, \tag{21}$$

  Here, $C_\psi = \int_0^{+\infty}\frac{|\hat{\psi}(\omega)|^2}{\omega}d\omega < +\infty$ serves as the normalization constant satisfying the wavelet admissibility condition, and the reproducing kernel $K(u_0, u, s_0, s) = \langle\psi_{u,s}, \psi_{u_0,s_0}\rangle$ measures the strong correlation between wavelet bases. Such redundancy can cause severe information leakage for the subsequent mask generator, hindering effective disentanglement of causal features.
- **Computational Intractability**: Dynamic graphs generally involve a massive number of nodes (where N is sufficiently large) and a long time span (T). Furthermore, it has been very widely recognized and also frequently observed that computing convolutions in the continuous domain not only incurs the curse of dimensionality but also inevitably requires evaluating infinite integrals.

To address these issues, we discretize the continuous scale, typically using exponential sampling (e.g., setting the scale $s = a^j$). Meanwhile, to avoid **boundary effects** and adapt the computational pipeline to deep learning tensor operations, we treat the node trajectories of dynamic graphs as periodic signals, thereby reformulating the wavelet transform as a **circular convolution**:

$$Wf[n, a^j] = \sum_{m=0}^{T-1} f[m]\psi_j^*[m-n] = f \circledast \bar{\psi}_j[n]. \tag{22}$$

Here, T denotes the total number of time steps in the input temporal trajectory; n and m are discrete time step indices; $f[m]$ represents the signal feature value at time step m; and a^j indicates the discrete scale parameter at the j-th level. Under this discrete formulation, we define the final constructed form of the core component in the main text, the **Discrete Complex Morlet Wavelet** is as follows:

$$\psi_s(t) = \underbrace{\exp\left(-\frac{1}{2}\left(\frac{t}{\sigma}\right)^2\right)}_{\text{Gaussian Envelope}} \cdot \underbrace{\exp\left(j \cdot \frac{\omega_0 t}{s}\right)}_{\text{Complex Oscillation}}. \tag{23}$$

The analyticity of this wavelet, guaranteed by the condition $\sigma^2\omega_0^2 \gg 1$, ensures that amplitude and phase are cleanly separated, which is theoretically beneficial for disentangling causal patterns from spurious transient shifts. To empirically verify that retaining the full complex coefficient is indeed necessary, we conduct a controlled ablation where the complex wavelet coefficients $\mathcal{W} \in \mathbb{C}^{N\times K\times T\times d}$ are replaced by their magnitudes $|\mathcal{W}| \in \mathbb{R}^{N\times K\times T\times d}$, while keeping all other components of WavCIL unchanged. Discarding ϕ via the magnitude operation removes temporal alignment information, which may hinder the model's ability to localize transient distribution shifts.

Tables 8 and 9 report the results on link prediction and node classification, respectively.

Table 8: Comparison of complex coefficients and magnitude-only coefficients on link prediction (AUC %). Best results are in bold.

Link Prediction (AUC %)							
	Real-world Datasets				**Synthetic Datasets**		
Variant	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
Magnitude-only	84.57±0.10	89.35±0.10	84.85±0.11	71.20±1.63	78.33±0.77	86.11±0.69	86.39±0.33
WavCIL (Complex)	**85.76±0.64**	**97.13±0.09**	**90.04±0.70**	**71.26±4.71**	**90.87±0.33**	**90.72±0.23**	**91.01±0.11**

Table 9: Comparison of complex coefficients and magnitude-only coefficients on node classification (ACC %). Best results are in bold.

Node Classification (ACC %)							
	Real-world Datasets				**Synthetic Datasets**		
Variant	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
Magnitude-only	47.25±0.99	50.71±0.24	42.53±0.49	47.27±5.01	30.33±0.97	28.32±1.35	27.66±0.63
WavCIL (Complex)	**51.88±0.76**	**53.78±0.41**	**45.57±1.01**	**58.53±0.68**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**

The results demonstrate that full complex coefficients substantially outperform magnitude-only representations, with particularly notable gains under extreme synthetic shifts. These empirical findings corroborate the theoretical analysis presented earlier: retaining phase information is critical for precisely localizing time-varying patterns and enabling the mask generator to reliably separate causal from spurious components, thereby validating the use of complex analytic wavelets in WavCIL.

B.2 Discrete Complex Wavelet Transform and Feature Mapping

Efficient computation based on FFT. To map the node temporal feature trajectories $\mathbf{Z} \in \mathbb{R}^{N\times T\times d}$ to the wavelet coefficient domain, direct computation of sliding window convolutions in the time domain would incur high computational complexity due to the large number of nodes N in dynamic graphs. Therefore, in the actual algorithmic implementation, we effectively leverage the convolution theorem to convert time-domain convolutions into the frequency-domain dot product, thereby reducing the complexity via the Fast Fourier Transform (FFT). Specifically, we now also precompute and then cache the conjugate-reversed FFT representations of the wavelet bases for all K scales:

$$\mathcal{F}(\psi_{s_k}) = \text{FFT}(\text{Flip}(\psi_{s_k}^*)). \tag{24}$$

Subsequently, we apply an orthonormal FFT to the input trajectories $\mathbf{Z}$ along the time dimension, then perform the batched dot product in the frequency domain via the broadcast mechanism, and finally transform back to the time domain using the Inverse Fast Fourier Transform (IFFT):

$$\mathbf{C}_{v,k} = \text{IFFT}\left(\text{FFT}(\mathbf{Z}_v) \odot \mathcal{F}(\psi_{s_k})\right) \in \mathbb{C}^{T\times d}. \tag{25}$$

Tail window pooling and boundary effects mitigation. Link prediction and node classification on dynamic graphs typically rely on the final state of the observation window. In order to mitigate the undesirable boundary effects that often occur in wavelet transform at the end of the sequence, rather than directly employing the coefficients of the last time step, we instead choose to extract a local window of length W from the end of the IFFT output (default set to $W = \min(3, T)$ in experiments) and also average along the time dimension to obtain the aggregated coefficients: [32]

$$\bar{\mathcal{W}}_{v,k} = \frac{1}{W} \sum_{t=T-W+1}^{T} \mathbf{C}_{v,k}^{t} \in \mathbb{C}^{d}. \tag{26}$$

The purpose of tail-window pooling is to summarize the localized wavelet responses immediately preceding the current forecast origin, while avoiding dependence on a single temporal coefficient. We clarify this design theoretically below.

For node v, scale s_k, and temporal location t, the DCWT coefficient is defined as:

$$C_{v,k,t} = \langle \mathbf{Z}_v, \psi_{s_k,t} \rangle , \tag{27}$$

where the wavelet basis $\psi_{s_k,t}$ is centered at temporal position t. Therefore, each coefficient already represents a temporally localized response.

For prediction at forecast origin r, the pooling operation in Equation (27) summarizes only the coefficients within the most recent local neighborhood:

$$\Omega_r = \{r - W + 1, \ldots, r\} , \tag{28}$$

according to:

$$\bar{\mathcal{W}}_{v,k}^{(r)} = \frac{1}{W} \sum_{t\in\Omega_r} \mathbf{C}_{v,k}^{t}. \tag{29}$$

The resulting $\bar{\mathcal{W}}_{v,k}^{(r)}$ remains conditioned on the forecast origin r and on the localized wavelet responses within Ω_r. Crucially, the pooling operation does not aggregate the entire temporal sequence and does not introduce information after the forecast origin, thus strictly respecting the temporal order of the forecasting task.

This design provides a balance between temporal specificity and local consistency. A window of $W = 1$ makes the representation depend on a single temporal-location coefficient, which may render it sensitive to short-lived fluctuations and noise. A moderately sized window combines nearby localized responses and captures whether a time-frequency pattern persists within the immediate forecasting context, thereby improving reliability while retaining temporal locality. If the window grows excessively large, however, responses from earlier temporal contexts may be mixed into the current representation, diluting the specificity of the local pattern and potentially introducing outdated information. We therefore use a short tail window, with $W = 3$ in the complete model and $W = \min(3, r)$ when fewer than three historical locations are available at the forecast origin.

Adaptive scale-domain interpolation. Since the preset number of computational scales may not match the feature channel dimension K_{len} required by downstream networks. To preserve the flexibility and compatibility of the framework, we separately extract the real and imaginary parts of the complex coefficients, align them to a fixed dimension of K_{len} via linear interpolation along the scale dimension, and finally output the wavelet coefficient tensor with the shape $\mathbb{C}^{N\times K_{len}\times d}$.

B.3 The Wavelet Coefficient-Domain Invariant Learning Module

After obtaining the multi-scale representation, the WavCIL framework employs an adaptive masking mechanism and environmental intervention regularization to enforce the model to disentangle causal patterns from spurious patterns within the coefficient domain, as illustrated in Figure 6.

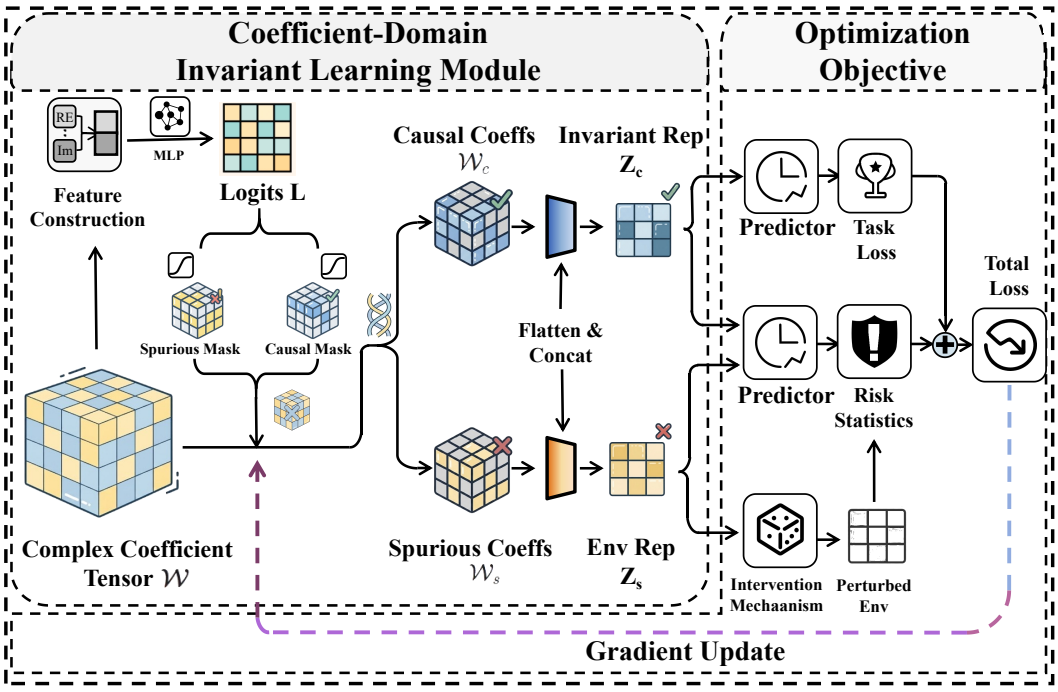

Figure 6: The pipeline of the Wavelet Coefficient-Domain Invariant Learning module.

Adaptive dual mask generation. Since analytic wavelets enable the separation of amplitude and phase components of a signal, the distinctions between causal patterns and spurious patterns are explicitly encoded in the real and imaginary parts of the wavelet coefficients. To leverage this information, we first extract the complex coefficients within the current observation window, and concatenate their real and imaginary parts along the feature dimension to construct the input feature $\mathbf{H}_{in} \in \mathbb{R}^{N\times(2Kd)}$ for the mask generator:

$$\mathbf{H}_{in} = \text{Concat}(\text{Re}(\mathcal{W}_v), \text{Im}(\mathcal{W}_v)). \tag{30}$$

Following the actual execution of these previous steps, a parameterized Multilayer Perceptron Network Φ_{mask} maps the given input to latent **importance logits** $\mathbf{L} \in \mathbb{R}^{N\times K}$:

$$\mathbf{L} = \Phi_{mask}\left(\mathbf{H}_{in}\right). \tag{31}$$

To achieve strict disentanglement, we employ a Sigmoid function with temperature scaling to generate two complementary masks—the **causal mask** $\mathbf{M}_c$ and the **spurious mask** $\mathbf{M}_s$:

$$\mathbf{M}_c = \sigma\left(\frac{\mathbf{L}}{\tau}\right), \quad \mathbf{M}_s = \sigma\left(-\frac{\mathbf{L}}{\tau}\right) = 1 - \mathbf{M}_c, \tag{32}$$

where $\sigma(\cdot)$ denotes the Sigmoid function and $\tau > 0$ is the temperature hyperparameter. The underlying design principle for this approach is found to consist of two distinct aspects:

- **Complementarity Constraint (Opposing Logits):** By applying the Sigmoid function to $\mathbf{L}$ and $-\mathbf{L}$ respectively, we enforce a strict complementary constraint (conceptually $\mathbf{M}_c + \mathbf{M}_s = 1$). This structure effectively drives the model to consistently make a clear binary decision for each wavelet coefficient: it either belongs to the invariant causal intervention or to the environmental noise, thereby also successfully preventing any information leakage or redundancy.
- **Discretization Approximation**: The temperature parameter τ controls the smoothness of the distribution. As $\tau \to 0$, the Sigmoid function approximates a step function, effectively driving the masks toward binarization $\{0, 1\}$. Furthermore, this design inherently encourages the model to make explicit "trade-off" decisions and eliminate ambiguous intermediate states, thereby also significantly enhancing the robustness and interpretability of the disentangled representations.

For downstream prediction tasks, it is essential to reconstruct the filtered multi-scale complex coefficients back to the real-valued feature space $\mathbb{R}^{N\times d}$. We consequently design a linear projection layer Φ_{proj} in order to flatten and further reduce the dimensionality of the reorganized coefficients, thereby obtaining both the causal representation $\mathbf{z}_c \in \mathbb{R}^{N\times d}$ that aggregates the stable time-frequency patterns and the spurious representation $\mathbf{z}_s \in \mathbb{R}^{N\times d}$ that captures non-stationary noise:

$$\mathbf{z}_c = \Phi_{proj}\left(\text{Flatten}[\text{Concat}(\text{Re}(\mathcal{W}_c), \text{Im}(\mathcal{W}_c))]\right), \tag{33}$$

$$\mathbf{z}_s = \Phi_{proj}\left(\text{Flatten}[\text{Concat}(\text{Re}(\mathcal{W}_s), \text{Im}(\mathcal{W}_s))]\right), \tag{34}$$

Optimization objective and intervention-based variance regularization. To guide model learning, we adopt an intervention optimization strategy based on risk variance. First, the standard task loss is effectively minimized under the already given causal representation:

$$\mathcal{L}_{task} = \mathbb{E}\left[\ell_{\text{CE}}(f_{pred}(\mathbf{z}_c), Y)\right]. \tag{35}$$

During training, to ensure $\mathbf{z}_c$ contains no environment features, we execute random environmental interventions. Specifically, we instead randomly permute the environmental representations within the same batch, and also mix the randomly sampled spurious confidence level $s = \sigma(y_s)$ with the causal prediction c_y to directly simulate the environmental distribution shift. We next recompute the loss under M random interventions and then calculate both its mean and its variance accordingly:

$$\mathcal{L}_{env} = \underbrace{\mathbb{E}[\ell(f(\mathbf{z}_c, \tilde{\mathbf{z}}_s), Y)]}_{\text{Risk Mean}} + \beta \cdot \underbrace{\text{Var}(\ell(f(\mathbf{z}_c, \tilde{\mathbf{z}}_s), Y))}_{\text{Risk Variance}}. \tag{36}$$

Here, minimizing the variance term forces the classifier f_{pred} to cut off the rely on $\tilde{\mathbf{z}}_s$.

To empirically verify the benefit of this variance regularization, we construct a degraded variant (extreme $\mathbf{Z}_s$) that uses a single spurious context per update, with the objective $\mathcal{L}_{env}^{single} = \widehat{R}_1$, thereby abandoning the variance penalty. Tables 10 and 11 compare this variant against the full WavCIL.

Table 10: Comparison of WavCIL and single-risk variant on link prediction (AUC %).

Link Prediction (AUC %)							
	Real-world Datasets				**Synthetic Datasets**		
Variant	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
WavCIL (extreme $\mathbf{Z}_s$)	84.51±0.06	89.28±0.26	84.81±0.07	66.39±1.07	72.45±4.56	81.25±1.09	81.63±0.21
WavCIL	**85.76±0.64**	**97.13±0.09**	**90.04±0.70**	**71.26±4.71**	**90.87±0.33**	**90.72±0.23**	**91.01±0.11**

Table 11: Comparison of WavCIL and single-risk variant on node classification (ACC %).

Node Classification (ACC %)							
	Real-world Datasets				**Synthetic Datasets**		
Variant	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
WavCIL (extreme $\mathbf{Z}_s$)	38.51±4.11	43.30±1.84	35.76±2.32	43.42±1.31	30.59±0.72	27.72±1.65	28.29±0.55
WavCIL	**51.88±0.76**	**53.78±0.41**	**45.57±1.01**	**58.53±0.68**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**

The full WavCIL consistently outperforms the single-risk variant across all datasets, with particularly substantial gains on Flight and under extreme synthetic shifts. These results confirm that relying on a single intervention samples only one spurious context, leaving the model vulnerable to variations. In contrast, multiple interventions combined with variance regularization cover a broader range of spurious patterns and explicitly penalize environment-sensitive predictions, thereby ensuring stronger OOD generalization.

The overall optimization objective is therefore defined as weighted sum of the two loss terms above:

$$\mathcal{L}_{total} = \underbrace{\mathcal{L}_{task}}_{\text{Eq. (35)}} + \lambda \cdot \underbrace{\mathcal{L}_{env}}_{\text{Eq. (36)}}. \tag{37}$$

By minimizing this overall objective, WavCIL can automatically suppress spurious correlations deeply coupled with specific contexts during training, thus maintaining precise prediction performance when facing unknown distribution shifts during the subsequent testing phase. Ultimately, WavCIL disentangles causal patterns from spurious patterns and enforces the model to rely exclusively on the causal patterns, thereby ensuring strong out-of-distribution generalization under distribution shifts.

In the actual backpropagation phase of the proposed algorithm, λ and β effectively regulate the penalty strength of invariance learning, thereby successfully helping us to achieve an optimal balance specifically between Empirical Risk Minimization and Out-of-Distribution Generalization.

C Additional Experiments

This section provides supplementary content for the experimental evaluations in the main text, and elaborates on the complete quantitative data of ablation studies, sensitivity analysis of key hyperparameters, as well as comparative results of the empirical time complexity for model training.

C.1 Quantitative Results of Ablation Studies

Section 4.4 in the main text presents a preliminary investigation into the contributions of core components. Tables 12 and 13 (shown below) list the quantitative results (AUC, ACC, and relative performance degradation Δ) of the full WavCIL and its three degraded variants across all datasets.

Table 12: Quantitative Results of the specific Ablation Study of our proposed WavCIL on multiple Link Prediction Tasks (AUC, where ↓ clearly indicates a performance drop).

Variant	Link Prediction											
	Wiki		Flight		Reddit		MOOC		Synthetic0.4		Synthetic0.6		Synthetic0.8	
	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ	AUC	Δ
WavCIL(Full)	85.76±0.64	-	97.13±0.09	-	90.04±0.70	-	71.26±4.71	-	90.87±0.33	-	90.72±0.23	-	91.01±0.11	-
w/o DCWT	79.92±1.69	↓5.84	95.34±1.70	↓1.79	87.95±0.92	↓2.09	68.64±10.86	↓2.62	77.11±0.98	↓13.76	72.56±1.10	↓18.16	71.22±0.63	↓19.79
w/o DM	83.54±0.11	↓2.22	91.35±0.98	↓5.78	89.02±0.14	↓1.02	73.33±1.61	↑2.07	85.66±0.87	↓5.21	85.39±0.24	↓5.33	85.08±0.04	↓5.93
w/o IL	85.82±0.25	↓0.06	92.60±0.97	↓4.53	89.87±0.20	↓0.17	72.35±3.27	↑1.09	89.43±0.57	↓1.44	89.73±0.24	↓0.99	89.64±0.09	↓1.01

Table 13: Quantitative Results of the specific Ablation Study of our proposed WavCIL on multiple Node Classification Tasks (ACC, where ↓ clearly indicates a performance drop).

Variant	Node Classification											
	Aminer15		Aminer16		Aminer17		OGBN-Arxiv		Synthetic0.4		Synthetic0.6		Synthetic0.8	
	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ	ACC	Δ
WavCIL(Full)	51.88±0.76	-	53.78±0.41	-	45.57±1.01	-	58.53±0.68	-	43.89±0.36	-	42.32±4.03	-	41.55±3.69	-
w/o DCWT	51.60±0.73	↓0.28	55.07±0.35	↑1.29	45.17±0.79	↓0.40	55.10±1.29	↓3.43	41.98±3.67	↓1.91	40.48±2.18	↓1.84	37.91±1.36	↓3.64
w/o DM	50.64±0.62	↓1.24	53.51±0.62	↓0.27	44.38±0.98	↓1.19	58.48±0.97	↓0.05	31.80±0.70	↓12.09	34.00±1.42	↓8.32	29.59±2.14	↓11.96
w/o IL	51.55±0.31	↓0.33	53.32±0.88	↓0.46	45.07±0.27	↓0.50	57.46±0.56	↓1.07	48.40±7.28	↑4.51	43.15±2.39	↓0.83	39.45±6.01	↓2.10

The data clearly reveal the following key observations:

- **The time-frequency localized representations are indispensable (Impact of DCWT):** Furthermore, compared with the full WavCIL model, the **w/o DCWT** variant actually suffers from significant performance drops across all datasets, especially under high-intensity distribution shifts (e.g., Synthetic 0.8) where the degradation is already particularly severe. This demonstrates the limitations of global spectral analysis: by losing localized resolution in the temporal dimension, the Fourier transform cannot precisely locate transient graph structural evolution. In contrast, the DCWT introduced in WavCIL provides both frequency and temporal information simultaneously, serving as the fundamental core for successfully capturing causal patterns.
- **The adaptive disentanglement is critical (Impact of DM):** The **w/o DM** variant also shows substantial performance degradation. Furthermore, this clearly indicates that simply transforming dynamic graph signals into the wavelet domain is already insufficient. Without actively filtering spurious components in specific environments via masking, causal patterns and spurious patterns actually remain consistently deeply entangled, causing the model to often easily fit biases of historical environments and thus completely fail to generalize when any test distribution shifts.
- **The invariance optimization ensures generalization capability (Impact of IL):** When the causal intervention and the variance penalty term are removed (**w/o IL**), the model's performance on the test set significantly declines. The full model enforces a reduction in the variance of

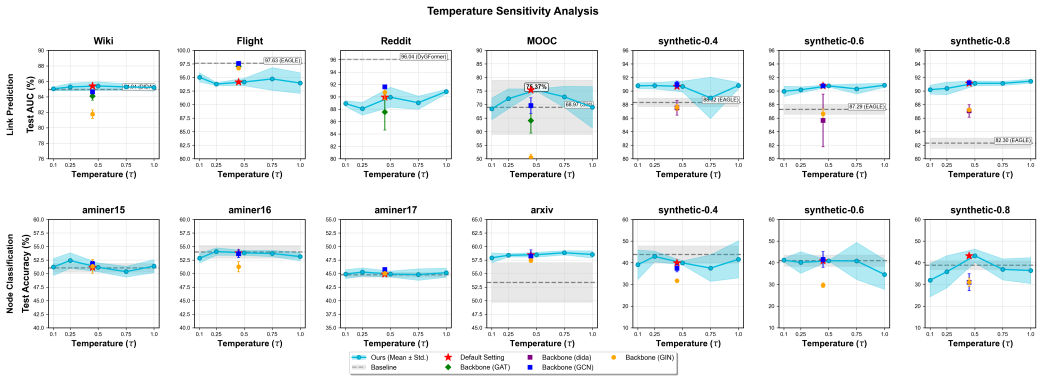

Figure 7: Visualization of the sensitivity analysis for the temperature parameter τ.

predictions across different latent environments by performing causal intervention on variant features. In contrast, the **w/o IL** variant lacks the mechanism to penalize prediction fluctuations caused by environmental changes, verifying that introducing environment-invariant constraints effectively guides the model to eliminate spurious correlations and achieve precise predictions.

C.2 Hyperparameter Sensitivity Analysis

To verify the anti-interference capability of the WavCIL framework, we evaluated the model's sensitivity to key hyperparameters and the backbone spatial encoder across multiple datasets.

Temperature Parameter τ: τ controls the degree of discretization in mask generation. In order to get a theoretically optimal balance between the "binarization purity" of masks and the "stability of gradient backpropagation," we therefore set the search range for τ to $[0.1, 1.0]$, covering transitions from smooth to steep. If $\tau \gg 1.0$, the mask degrades into nearly uniform/flat soft attention, losing the ability to enforce disentanglement between causal and spurious patterns. WavCIL consistently exhibits high stability across the wide range of $\tau \in [0.1, 1.0]$, as shown in Figure 7 & Table 14. When τ is close to 0.25, the model actually achieves peak performance on most datasets. This aligns with our theoretical expectations in Section B.3: a moderate temperature not only ensures effective gradient backpropagation but also encourages the model to make explicit binarized disentangle decisions.

Number of Scales K**:** To explore the model's coverage requirements for temporal frequency bands in dynamic graphs, we follow the Sampling Theorem and test wavelet scale numbers $K \in 4, 8, 12, 16, 32$ with multiplicative spans. This interval was designed to fully cover the evolving feature space, ranging from coarse-grained low-frequency features (which may lead to under-representation) to fine-grained high-frequency features (which may approach the edge of over-parameterization). As K increases, the model's ability to capture multi-scale frequency information strengthens, with performance steadily improving and reaching optimal levels in the range $K \in [8, 16]$. When $K = 32$, a slight performance drop is observed on some datasets, likely due to the introduction of high-frequency redundant noise or parameter overfitting, as illustrated in Figure 8 & Table 15.

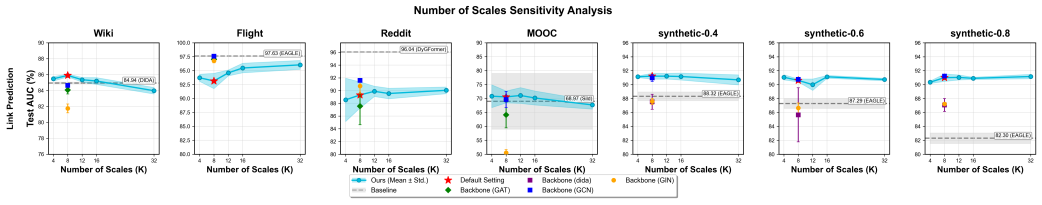

Figure 8: Visualization of the sensitivity analysis for the number of wavelet scales K.

Center Frequency ω_0**:** ω_0 determines the complex oscillation characteristics of the wavelet basis. According to the admissibility condition for the continuous complex Morlet wavelet in Section 3.1, the center frequency must satisfy $\omega_0 \geq 5$ (with an empirical value of 6 typically used) to ensure that the negative frequency components of the Gaussian window function are strictly near zero, thereby making the basis function a analytic wavelet capable of separating amplitude and phase. Accordingly,

Table 14: Sensitivity Analysis of Temperature (τ) across multiple datasets.

Sensitivity Analysis of Temperature (τ)

Link Prediction (AUC %)

	Real-world Datasets				Synthetic Datasets		
τ	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
0.100	85.04±0.25	**95.04±0.84**	88.92±0.47	68.46±4.00	90.78±0.44	89.97±0.77	90.21±0.65
0.125	84.60±0.35	93.42±0.07	84.57±0.08	70.55±4.97	84.72±0.88	86.53±0.75	86.24±0.58
0.250	85.28±0.49	93.79±0.28	88.08±1.03	72.18±3.65	**90.81±0.48**	90.19±0.44	90.39±0.93
0.375	84.35±0.28	90.47±0.93	84.62±0.18	69.11±6.34	77.65±0.67	86.25±0.60	86.36±0.32
0.500	**85.40±0.54**	94.17±0.68	89.95±1.61	**75.37±0.40**	90.67±0.80	90.76±0.17	91.14±0.35
0.625	84.49±0.11	89.15±0.06	84.89±0.24	67.11±6.56	78.63±1.31	85.86±0.50	86.16±0.56
0.750	85.28±0.41	94.74±2.10	89.05±1.07	72.86±3.91	88.99±3.05	90.32±4.57	91.14±0.31
0.875	84.88±0.32	89.00±0.25	84.96±0.06	68.95±5.10	78.61±0.36	85.53±0.25	86.05±0.49
1.000	85.22±0.48	93.98±1.91	**90.86±0.36**	69.02±7.64	90.81±0.48	**90.86±0.30**	**91.45±0.28**

Node Classification (ACC %)

	Real-world Datasets				Synthetic Datasets		
τ	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
0.100	51.22±1.57	52.85±0.88	44.91±0.82	57.92±0.92	39.11±6.79	**41.22±0.91**	31.96±7.65
0.125	36.90±0.80	42.81±0.75	34.82±0.68	28.24±11.62	26.52±2.62	24.55±2.65	25.99±1.25
0.250	**52.42±1.43**	**54.08±0.69**	**45.30±0.76**	58.39±0.30	**42.92±2.46**	40.15±4.98	35.87±7.40
0.375	33.45±4.56	38.91±3.35	31.33±3.80	32.17±0.38	25.38±1.46	24.75±1.06	26.62±0.31
0.500	51.17±0.82	53.83±0.45	44.92±0.55	58.49±0.44	39.95±0.93	40.92±0.15	**43.25±3.06**
0.625	34.40±2.61	39.90±2.64	31.94±1.97	25.55±12.40	25.89±0.90	24.01±2.51	26.59±1.49
0.750	50.37±1.00	53.72±0.58	44.83±1.06	**58.85±0.40**	37.54±6.11	40.85±8.57	36.97±4.85
0.875	36.04±2.84	43.49±1.52	34.74±1.42	30.65±2.89	26.19±0.38	25.15±1.93	26.52±1.28
1.000	51.38±1.20	53.15±0.77	45.17±0.84	58.54±0.66	41.62±8.58	34.60±6.85	36.41±5.97

Table 15: Sensitivity Analysis of Number of Scales (K) across multiple datasets.

Sensitivity Analysis of Number of Scales (K)

Link Prediction (AUC %)

	Real-world Datasets				Synthetic Datasets		
K	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
4	85.48±0.28	93.73±0.64	88.54±3.40	70.84±4.09	91.13±0.12	91.05±0.37	90.34±0.17
8	**85.90±0.18**	93.15±1.37	89.31±2.05	70.58±2.14	**91.21±0.43**	90.60±0.21	91.00±0.54
12	85.36±0.29	94.60±0.53	89.88±0.81	**71.09±2.83**	**91.21±0.41**	89.99±0.82	91.03±0.36
16	85.20±0.43	95.48±0.86	89.54±0.83	70.20±2.51	91.16±0.50	**91.12±0.20**	90.90±0.16
32	83.98±0.41	**96.04±0.80**	**90.04±0.53**	67.70±1.50	90.68±0.72	90.73±0.18	**91.16±0.32**

we adopted $[6, 10]$ that strictly satisfy the boundaries of mathematical theory as the test interval. Across tests with $\omega_0 \in [6, 10]$, model performance shows minimal fluctuation, with variance typically below 1%, demonstrating that the frequency band energy distribution of the discrete complex wavelet basis is not critically dependent on the choice of base frequency, as shown in Figure 9 & Table 16.

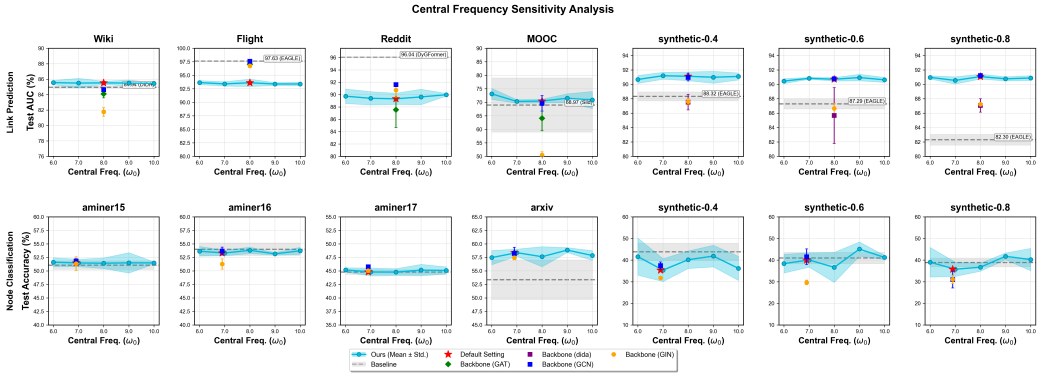

Figure 9: Visualization of the sensitivity analysis for the central frequency ω_0.

Table 16: Sensitivity Analysis of Central Frequency (ω_0) across multiple datasets.

Sensitivity Analysis of Central Frequency (ω_0)

Link Prediction (AUC %)

	Real-world Datasets				Synthetic Datasets		
ω_0	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
6	**85.55±0.28**	**93.65±0.37**	89.74±1.18	**73.13±2.03**	90.67±0.55	90.45±0.35	90.96±0.22
7	85.49±0.62	93.41±0.40	89.40±1.07	70.32±0.65	**91.19±0.48**	90.83±0.13	90.54±0.50
8	85.52±0.21	93.62±0.68	89.32±0.91	70.47±0.67	91.11±0.60	90.70±0.24	**91.07±0.34**
9	85.49±0.40	93.39±0.27	89.62±1.25	71.52±1.62	90.98±0.83	**90.94±0.45**	90.76±0.29
10	85.45±0.12	93.39±0.32	**89.98±0.22**	70.86±3.17	91.09±0.49	90.62±0.35	90.87±0.33

Node Classification (ACC %)

	Real-world Datasets				Synthetic Datasets		
ω_0	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
6	**51.61±0.82**	53.64±0.89	**45.19±0.39**	57.47±1.32	41.62±8.58	38.38±4.34	39.04±6.86
7	51.47±0.68	53.31±0.51	44.80±0.74	58.37±0.66	35.37±5.27	40.05±3.32	35.77±3.31
8	51.42±0.97	**53.81±0.59**	44.78±0.63	57.63±1.86	40.18±4.07	36.64±6.94	36.71±1.86
9	51.49±1.89	53.16±0.23	45.15±1.07	**58.85±0.46**	**41.85±4.99**	**45.09±3.31**	**41.75±1.56**
10	51.48±0.19	53.72±0.65	45.08±0.66	57.85±0.90	36.17±5.56	41.22±0.91	40.25±5.17

Backbone Sensitivity: To validate the generality of the framework, we replace the underlying graph spatial encoder of WavCIL with several standard architectures, including GAT, GCN, and GIN. Experimental results show that across both link prediction and node classification tasks, WavCIL equipped with different backbones consistently delivers stable and highly competitive performance, as illustrated in Table 17. For instance, on the Flight dataset, variants based on GCN and GAT achieve excellent AUC scores of 97.58% and 97.04%, respectively. On synthetic datasets with extreme distribution shifts (Shift Level 0.8), variants such as those based on GCN and GIN maintain strong anti-interference capability and significantly outperform most baseline models. This strongly confirms that WavCIL is not deeply entangled to a specific graph neural network architecture, but rather serves as a general-purpose framework that can be seamlessly integrated as a plug-and-play module into various spatial encoders, endowing them with powerful OOD generalization capabilities.

Table 17: Performance of WavCIL using various standard spatial encoders as backbones. (Note: For Synthetic Link Prediction datasets, DIDA performance is reported in the first row instead of GAT).

Backbone Sensitivity

Link Prediction (AUC %)

	Real-world Datasets				Synthetic Datasets		
Backbone	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
WavCIL (GAT) / (DIDA)	84.08±0.51	97.04±0.18	87.55±2.91	64.11±4.64	87.53±1.08	85.66±3.89	87.06±0.93
WavCIL (GCN)	84.64±0.13	97.58±0.04	91.60±0.14	69.64±2.90	90.96±0.50	90.76±0.25	91.22±0.12
WavCIL (GIN)	81.76±0.56	96.76±0.36	90.72±1.13	50.52±1.19	87.58±0.45	86.64±0.72	87.22±0.11

Node Classification (ACC %)

	Real-world Datasets				Synthetic Datasets		
Backbone	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
WavCIL (GCN)	51.81±0.46	53.68±0.71	45.76±0.14	58.23±1.15	37.47±1.66	41.55±3.64	31.00±3.91
WavCIL (GIN)	51.35±1.26	51.28±0.95	45.05±0.38	57.45±0.41	31.73±0.45	29.66±1.04	31.06±1.95

Tail-window Size W: The tail-window pooling in Eq. (26) aggregates wavelet coefficients over the last W time steps to summarize localized wavelet responses immediately preceding the forecast origin, while avoiding dependence on a single temporal coefficient. This design provides a balance between temporal specificity and local consistency. We evaluate $W \in \{1, 2, 3, 4, 5\}$ while keeping all other settings unchanged. Table 18 report the results on link prediction and node classification. Among the tested windows, $W = 3$ achieves the best performance on 13 out of 14 task–dataset pairs. $W = 1$ is generally weaker, especially on Flight and synthetic datasets, confirming that relying on a single temporal coefficient fails to effectively characterize distribution shifts driven by local events or phased variations. As W increases beyond 3, performance tends to degrade, indicating that excessively large windows dilute the specificity of the localized pattern by mixing information from earlier temporal contexts. These results support $W = 3$ as the default configuration.

Table 18: Sensitivity Analysis of Tail-Window Size (W) across multiple datasets.

Sensitivity Analysis of Tail-Window Size (W)							
Link Prediction (AUC %)							
	Real-world Datasets				**Synthetic Datasets**		
W	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
1	84.54±0.22	89.07±1.82	84.74±0.10	71.08±5.99	79.07±1.47	84.31±0.14	84.43±0.26
2	84.62±0.04	89.96±0.82	84.76±0.09	**71.34±1.44**	79.91±1.79	86.16±0.40	85.95±0.28
3	**85.76±0.64**	**97.13±0.09**	**90.04±0.70**	71.26±4.71	**90.87±0.33**	**90.72±0.23**	**91.01±0.11**
4	84.17±0.50	89.59±0.03	84.97±0.23	70.81±1.17	79.46±1.78	85.62±0.54	85.97±0.46
5	84.10±0.09	89.52±0.10	85.08±0.11	68.52±3.50	80.54±1.11	85.11±0.84	84.95±0.42
Node Classification (ACC %)							
	Real-world Datasets				**Synthetic Datasets**		
W	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
1	48.63±1.95	51.48±0.63	43.33±0.27	51.98±2.18	28.59±0.45	26.62±2.12	27.79±1.17
2	47.34±1.09	50.79±1.53	42.67±1.06	51.38±2.50	28.69±1.00	26.05±2.63	27.02±1.13
3	**51.88±0.76**	**53.78±0.41**	**45.57±1.01**	**58.53±0.68**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**
4	47.90±1.92	51.36±0.80	42.94±1.01	51.24±3.57	29.09±0.88	25.48±1.50	28.69±1.29
5	49.56±1.47	52.18±0.95	43.68±0.71	52.22±4.24	30.09±1.01	25.75±0.66	27.39±1.43

C.3 Comparison with Spectral Baselines

The current manuscript already contains Fourier-based spectral invariant learning and other time-frequency transforms. To further substantiate the claimed advantage of WavCIL, we consolidate a comprehensive comparison with three representative spectral baselines: SILD represents global Fourier invariant learning; DEFT-T represents learnable graph-wavelet learning; and Strictly-Aligned FFT uses the same encoder, masks, projections, losses, optimization, model-selection protocol, and capacity as WavCIL, changing principally DCWT to FFT. For clarity, we consolidate their complete results in tables 19 and 20.

Table 19: Consolidated comparison with spectral baselines on dynamic link prediction (AUC %). The best results are in bold, second-best underlined.

Link Prediction (AUC %)							
	Real-world Datasets				**Synthetic Datasets**		
Method	**Wiki**	**Flight**	**Reddit**	**MOOC**	**0.4**	**0.6**	**0.8**
DEFT-T	54.01±1.26	61.73±4.60	78.88±13.11	63.75±6.32	73.25±3.47	60.77±3.91	56.06±6.07
SILD	80.02±0.60	91.98±1.26	86.73±1.68	68.97±10.00	85.96±0.19	84.64±1.25	78.01±0.70
Strictly-Aligned FFT	72.31±2.03	96.59±0.09	85.69±2.07	69.53±1.19	84.32±1.50	80.02±0.74	78.51±2.14
WavCIL (Ours)	**85.76±0.64**	**97.13±0.09**	**90.04±0.70**	**71.26±4.71**	**90.87±0.33**	**90.72±0.23**	**91.01±0.11**

Table 20: Consolidated comparison with spectral baselines on temporal node classification (ACC %). Best results are in bold, second-best underlined.

Node Classification (ACC %)							
	Real-world Datasets				**Synthetic Datasets**		
Method	**Aminer15**	**Aminer16**	**Aminer17**	**Arxiv**	**0.4**	**0.6**	**0.8**
DEFT-T	44.20±5.14	46.21±4.32	36.85±4.66	14.99±0.07	43.33±2.11	40.00±4.16	40.90±5.37
SILD	51.06±0.80	**54.00±1.16**	44.77±0.43	53.36±3.65	43.82±3.97	40.95±2.64	38.88±2.00
Strictly-Aligned FFT	47.24±1.19	49.12±1.91	40.43±1.04	56.07±2.41	39.38±3.56	32.06±2.17	34.47±0.30
WavCIL (Ours)	**51.88±0.76**	53.78±0.41	**45.57±1.01**	**58.53±0.68**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**

Overall, these experiments strongly support our claim that WavCIL's superior OOD generalization stems from the joint time-frequency localization of the discrete complex wavelet transform, which enables fine-grained disentanglement of causal and spurious patterns in dynamic graphs.

C.4 Comparison with Recent Dynamic Graph OOD Methods

To provide a more up-to-date and comprehensive empirical evaluation, we expand the baseline set with recent dynamic graph OOD generalization methods that have publicly available code or can

be reasonably adapted to our benchmark. Specifically, we include **DyCIL** [33] , **EvoGOOD** [34] , **DyAug** [35] , **EQuAD** [36] , and **FreeDyG** [37] . All methods are evaluated under the same data partitions, tasks, metrics, and three-seed random protocol as in the main experiments.

Tables 21 and 22 report the full comparison on link prediction and node classification, respectively. For completeness, we also include the results of WavCIL.

Table 21: Performance comparison with recent dynamic graph OOD methods on link prediction tasks (AUC %). The best results are clearly highlighted in bold, and the second-best are underlined.

	Real-world Datasets (AUC %)					Synthetic Datasets (AUC %)				Overall (AUC %)	
Model	**Wiki**	**Flight**	**Reddit**	**MOOC**	**Average**	**0.4**	**0.6**	**0.8**	**Average**	**Average**	**Δ**
Recent Dynamic Graph OOD Methods											
DyCIL	80.52±0.20	**97.57±0.01**	85.56±0.01	55.46±0.57	79.78	89.93±0.12	85.56±0.01	79.99±0.17	85.16	82.08	+6.75
EvoGOOD	81.96±0.53	91.86±0.65	86.89±2.20	68.26±3.27	82.24	90.84±0.52	89.21±0.85	90.32±0.31	90.12	85.62	+3.21
DyAug	79.14±0.70	95.86±1.05	81.53±1.34	54.84±0.83	77.84	77.57±0.38	76.32±0.26	76.70±0.15	76.86	77.42	+11.41
EQuAD	79.60±0.34	97.55±0.03	88.54±0.18	**86.11±0.13**	**87.95**	**92.72±0.07**	89.62±0.05	84.91±0.13	89.08	88.44	+0.39
FreeDyG	82.61±1.54	95.08±0.33	88.35±2.87	50.86±0.89	79.23	60.92±9.46	61.27±9.77	61.41±9.89	61.20	71.50	+17.33
Our Method											
WavCIL (Ours)	**85.76±0.64**	97.13±0.09	**90.04±0.70**	71.26±4.71	86.05	90.87±0.33	**90.72±0.23**	**91.01±0.11**	**90.87**	**88.83**	-

Table 22: Performance comparison with recent dynamic graph OOD methods on node classification tasks (ACC %). The best results are clearly highlighted in bold, and the second-best are underlined.

	Real-world Datasets (ACC %)					Synthetic Datasets (ACC %)				Overall (ACC %)	
Model	**Aminer15**	**Aminer16**	**Aminer17**	**OGBN-Arxiv**	**Average**	**0.4**	**0.6**	**0.8**	**Average**	**Average**	**Δ**
Recent Dynamic Graph OOD Methods											
DyCIL	51.63±0.94	53.62±0.47	44.94±0.64	58.13±0.12	52.08	21.04±2.22	21.01±1.04	24.25±4.84	22.10	39.23	+8.99
EvoGOOD	15.64±1.69	22.25±0.66	17.14±1.11	15.01±1.22	17.51	24.98±1.72	24.22±1.26	25.89±1.72	25.03	20.73	+27.49
DyAug	50.28±0.94	53.60±0.83	44.24±0.29	**60.40±0.92**	52.13	22.91±1.95	26.29±5.65	30.26±2.81	26.49	41.14	+7.08
EQuAD	48.69±0.42	51.56±0.51	43.56±0.23	57.09±0.68	50.23	26.35±1.06	23.71±0.47	27.89±0.35	25.98	39.83	+8.38
FreeDyG	7.28±6.63	12.12±7.55	9.24±5.13	12.38±7.35	10.26	19.97±0.65	20.98±0.49	20.01±1.07	20.32	14.57	+33.65
Our Method											
WavCIL (Ours)	**51.88±0.76**	**53.78±0.41**	**45.57±1.01**	58.53±0.68	**52.44**	**43.89±0.36**	**42.32±4.03**	**41.55±3.69**	**42.59**	**48.22**	-

WavCIL achieves the best mean on 10/14 settings: four link-prediction and six node-classification datasets. It outperforms DyCIL and DyAug on 13/14 settings and EvoGOOD and FreeDyG on all 14. EQuAD is stronger on Flight, MOOC, and Syn. 0.4 link prediction; DyCIL is best on Flight, while DyAug is best on Arxiv. These complementary results are retained, while WavCIL shows more consistent advantages on node classification.

D Implementation Details

D.1 Task Definitions

To comprehensively evaluate the generalization capability of the WavCIL framework in dynamic graph settings under Out-of-Distribution (OOD) shifts, we construct experimental setups by employing two of the most representative downstream tasks for dynamic graph representation learning:

- **Dynamic Link Prediction**: This task aims to assess the model's ability to capture the complex evolutionary dynamics of graph topology. Specifically, given a sequence of graph snapshots within a historical time window $\mathcal{G}_{\leq t}$, the model is required to predict the probability that a new interactive edge emerges between any given target node pair (u, v) in the next time step $t + 1$.
- **Temporal Node Classification**: This task aims to evaluate the model's reasoning capability regarding node semantic features in a non-stationary evolving environment under unknown distribution shifts. Given the structural evolution and node feature trajectories of the dynamic graph, the model needs to predict the category label of a node at a specific future time step in the unseen test environment(e.g., the publication venue or research field of an academic paper).

D.2 Evaluation Metrics

For the two tasks described above, we respectively adopt ROC-AUC and Accuracy as the core evaluation metrics to rigorously evaluate the proposed model under unknown distribution shifts.

ROC-AUC (Area Under the ROC Curve) In dynamic link prediction, the number of node pairs that will interact in the future (positive samples) is far smaller than those that will not (negative samples). Therefore, we use ROC-AUC as the primary metric to robustly handle this extreme class imbalance problem. During evaluation, we randomly sample negative samples from the unobserved node pairs while keeping the numbers of positive and negative samples same. AUC measures the probability that the model ranks a true positive sample ahead of a randomly selected negative sample. Formally, let $\mathcal{P}$ be the set of positive edges and $\mathcal{N}$ the set of negative edges; let p_i and p_j be the predicted probabilities for a positive sample and a negative sample, respectively. Then AUC is computed as:

$$AUC = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{i \in \mathcal{P}} \sum_{j \in \mathcal{N}} \mathbb{I}(p_i > p_j). \tag{38}$$

A high AUC value indicates that the model remains capable of accurately capturing the underlying causal mechanisms driving node interactions, even under severe distribution shifts, thereby demonstrating superior ranking and discriminative ability across all unknown test environments.

Accuracy (ACC). In the temporal node classification task, we use accuracy (ACC) to quantitatively measure the overall prediction precision of the proposed model under highly complex distribution shifts. ACC quantifies the proportion of node category labels correctly predicted by the model in unseen test environments (e.g., future year sequences). Formally, given the total number of test nodes N_{test}, where the ground-truth label of node i is y_i and the predicted label is $\hat{y}_i$, ACC is defined as:

$$ACC = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \mathbb{I}(\hat{y}_i = y_i), \tag{39}$$

where $\mathbb{I}(\cdot)$ is the standard indicator function, which equals 1 if and only if the predicted label exactly matches the ground-truth label, and 0 otherwise, strictly penalizing any incorrect model predictions.

D.3 Datasets Construction and OOD Splitting Details

To validate the anti-interference capability of our model against different types of complex environmental shifts, we conduct extensive experiments on six real-world datasets and two controlled synthetic datasets with varying levels of injected spurious correlation intensity. The feature processing, scale statistics, and temporal OOD splitting strategies for each dataset are elaborated as follows:

Table 23: Statistics of the datasets used in our experiments.

Dataset	Wiki	Flight	Reddit	MOOC	Aminer	Arxiv	Synthetic-LP	Synthetic-NC
Steps	16	16	16	16	17	13	16	100
Nodes	9,227	12,257	10,984	7,144	43,141	36,362	23,035	5,000
Edges	34,203	208,919	241,332	226,608	827,527	66,176	151,790	11,139,978
Dims	64	32	64	64	128	128	64	4

Real-world link prediction datasets: In such tasks, the distribution shifts primarily arise from users' interest drift or sudden event impacts, which naturally occur as real-world systems evolve over time.

- **Wiki**[13]: Constructed from Wikipedia page editing records over one month. Nodes represent editors and Wiki pages, and edges correspond to timestamped edit requests. Since the raw data does not provide node features, we first aggregate the 172-dimensional LIWC edge feature vectors extracted from edit texts and map them into 63-dimensional node representations. A 1-dimensional node type indicator is further introduced to explicitly distinguish users from pages, resulting in 64-dimensional node features. The continuous time axis is divided into 16 deduplicated graph snapshots, which are strictly split chronologically into training / validation / test sets with a ratio of 10:1:5. The dataset contains 9,227 nodes and 34,203 edges.
- **Flight**[14]: Built upon the real flight operation records from tgbl-flight-v2, where nodes represent airports and edges represent flight connections. To control the scale, we sample every 200 records according to the global row index. Node features are encoded as 32-dimensional vectors based on airport physical attributes (type, continent, region, latitude and longitude). The time axis is

divided into 16 snapshots, split chronologically with a ratio of 10:1:5 to evaluate generalization under temporal distribution shifts. The dataset contains 12,257 nodes and 208,919 edges.

- **Reddit**[13]: Models posting behaviour on subreddits over one month. Nodes represent users and posts, and edges represent posting requests. We aggregate the 172-dimensional LIWC edge features, map them into 63-dimensional node representations, and add a 1-dimensional type indicator to form 64-dimensional node features. The temporal snapshot division and data splitting logic are exactly the same as above. The dataset contains 10,984 nodes and 241,332 edges.
- **MOOC**[13]: A student interaction network built from online course content. As the raw data does not contain node features, we map the 4-dimensional raw edge features into a high-dimensional representation, then aggregate outgoing edge features for user nodes and incoming edge features for course nodes, and finally obtain 64-dimensional features by combining 1-dimensional node type identifiers for semantic distinction. The temporal snapshot division and data splitting logic are exactly the same as above. The dataset contains 7,144 nodes and 226,608 edges.

Real-world node classification datasets: The observed distribution shifts mainly arise from long-term changes in academic research hotspots and cross-year evolution of citation patterns.

- **Aminer**[15, 16]: A scholarly citation network dataset where the node label is the publication venue of a paper. We adopt a strict chronological split to construct OOD scenarios: papers published from 2001 to 2011 are used for training, papers from 2012 to 2014 for validation, and papers published since 2015 for testing. The model is required to effectively leverage complex historical dynamic graph structures to make precise predictions for unseen future years.
- **OGBN-Arxiv**[17, 18]: A citation network of computer science papers. Each node is associated with a dense 128-dimensional semantic feature (composed of the average word embeddings from the paper titles and abstracts) and its publication year. We sample the categories, retaining 20 classes to build the downstream classification task. The temporal split is performed strictly chronologically by year: papers published in 2018 and earlier are used for training, those published in 2019 for validation, and those published in 2020 and later for unseen future testing.

Synthetic datasets with controllable distribution shifts: These datasets are used to quantitatively evaluate the model's ultimate feature disentanglement capability under specific interference.

- **Synthetic (Link Prediction)**[5]: Built upon the Collab dataset. On top of the original network structure, we generate additional variable features for each node at each timestamp. These extra features are artificially designed to have a spurious correlation with the future link labels. The shift level is set to 0.4, 0.6, and 0.8 during the training and validation phases, while it is forcibly set to 0 during the testing phase, in order to strictly examine whether the proposed model successfully avoids relying on these injected spurious correlation patterns for making predictions.
- **Synthetic (Node Classification)**[5]: Generated by a Stochastic Block Model (SBM). The edge probability at each step is jointly determined by two frequency factors: one remains perfectly correlated with the class label all the time (invariant factor); the other spurious factor whose correlation with labels changes sharply across different shift levels (0.4, 0.6, 0.8). The shift level for the test set is also forcibly set to 0 to rigorously assess true invariant feature learning.

D.4 Baseline Details

To comprehensively verify the generalization superiority of the proposed paradigm, we compare WavCIL with 11 representative baseline models across three major categories to highlight its effectiveness. The core architecture and underlying mechanism of each model are described as follows:

Generic dynamic graph models: These models represent a superior top level of spatio-temporal graph neural networks and have strong evolutionary modeling capabilities. However, it is clearly the case that they are still routinely built on the independent and identically distributed (I.I.D.) assumption and also presently lack any active intervention or disentanglement mechanisms to cope with the highly complex environmental shifts that are arising from unknown distribution changes.

- **DyGFormer**[20]: Efficiently captures spatio-temporal interactions in dynamic graphs by exploring a neighbor co-occurrence mechanism. It uses a patching technique to group historical interaction sequences and employs a Transformer architecture for global attention modeling.

- **GraphMixer**[21]: This model adopts an MLP-Mixer architecture to replace traditional complex attention mechanisms for capturing temporal link evolution. It aggregates historical neighbors within fixed-length time windows, resulting in a dynamic graph model that offers both high computational efficiency and strong representational power under standard experimental settings.
- **TGN (Temporal Graph Networks)**[23]: A classic continuous-time dynamic graph model based on dynamic memory modules. The TGN framework maintains and recursively updates an implicit memory state for each node. Although it performs well on stationary streaming graph data, it becomes highly susceptible to irreversibly accumulating non-stationary noise from the historical environment into its memory units when facing unseen temporal distribution shifts.
- **DEFT-T (Dynamic Wavelets for Global Interactions with Transformer)**[22]: An advanced framework that extends dynamic graph representation learning to the spectral domain. To overcome the drawback that traditional spatial aggregation only captures local low-frequency signals, DEFT learns time-evolving spectral graph wavelets via parameterized Chebyshev polynomial approximation, so as to more effectively capture complex global high-frequency interactions in dynamic graphs. DEFT-T is the optimal variant of this framework, which uses a sparse Transformer (self-attention mechanism) to fuse spatial local features with wavelet global features. Nevertheless, although it leverages wavelet analysis to enhance the representation power of graph representations, it is originally designed primarily to improve prediction performance on standard I.I.D data. The framework does not explicitly incorporate causal disentanglement or invariant learning mechanisms, leaving it still limited in combating unseen temporal OOD shifts.

Generic OOD methods: These methods originate from classical machine learning. Furthermore, it is also the case that in our own experiments, we directly incorporate these optimization strategies into dynamic graph backbones (e.g., GAT or GCN), specifically in order to actually evaluate the limitations of pure optimization-level constraints under complex temporal distribution shifts.

- **IRM (Invariant Risk Minimization)**[24]: Aims to learn a robust data representation such that the optimal classifier built on top of this representation performs well across all training environments simultaneously. In implementation, it enforces constraints by adding a squared norm penalty on the gradients of virtual classifiers to the objective of empirical risk minimization.
- **V-REx (Variance Risk Extrapolation)**[25]: Based on the intuition that the errors of a robust model should be similar across different environments. It reduces the model's overfitting sensitivity to specific environments by extra penalizing (minimizing) the variance of empirical risks across different training environments to mitigate spurious environmental correlations.
- **GroupDRO (Group Distributionally Robust Optimization)**[26]: Abandons optimizing the average loss over the entire dataset and instead directly optimizes the worst-case loss across all known training groups (environments), thereby improving the model's performance lower bound under extreme shifts by effectively mitigating the adverse impact of distribution changes.

Dynamic graph OOD methods: Furthermore, it is also widely acknowledged that these models represent the most advanced state-of-the-art explorations specifically dedicated to addressing complex and unpredictable temporal distribution shifts within the provided dynamic graph data.

- **DIDA (Disentangled Interventional Dynamic Graph Attention)**[1]: As a representative of spatio-temporal invariant learning, DIDA uses an environment inference network to capture latent non-stationary temporal patterns and employs a disentangled spatio-temporal attention mechanism to separate causal and spurious features in the original spatio-temporal domain.
- **SILD (Spectral Invariant Learning on Dynamic Graphs)**[5]: The current state-of-the-art representative of spectral-domain invariant learning. SILD maps temporal features entirely into the frequency domain using a global Fourier transform and attempts to separate high-frequency and low-frequency features via static frequency filters. It serves as the most critical benchmark for demonstrating that "local time-frequency analysis (wavelet transform) is superior to global spectral analysis (Fourier transform)" within our proposed WavCIL framework.
- **EAGLE**[4]: This model investigates dynamic graph OOD generalization from an environment learning perspective. It models latent environments through an environment-aware DGNN and integrates environment instantiation, invariant pattern recognition, and fine-grained causal intervention to enhance generalization capabilities against temporal distribution shifts.

- **STONE**[27]: This model is a learning framework tailored for spatio-temporal out-of-distribution (OOD) challenges, initially designed for traffic prediction. It enhances robustness against spatio-temporal distribution shifts by constructing spatial and temporal semantic graphs, alongside the utilization of Fréchet embeddings, graph interventions, and an explore-to-extrapolate objective.

All the experiments are conducted on an NVIDIA GeForce RTX 3090 GPUs with 24GB of memory.

E Case Study

To intuitively validate the disentanglement capability of WavCIL in addressing real-world distribution shifts, we present a visualization case built on the MOOC[13] dataset in Figure 10. In this setting, nodes represent students and course resources, and edges denote time-evolving interactive behaviors. At a specific time step T_4, the network undergoes a local, sudden aggregation of interactions(This may correspond to distribution shifts induced by external contextual perturbations, e.g., finals weeks.). This reflects the complex and time-varying nature of real-world distribution shifts.

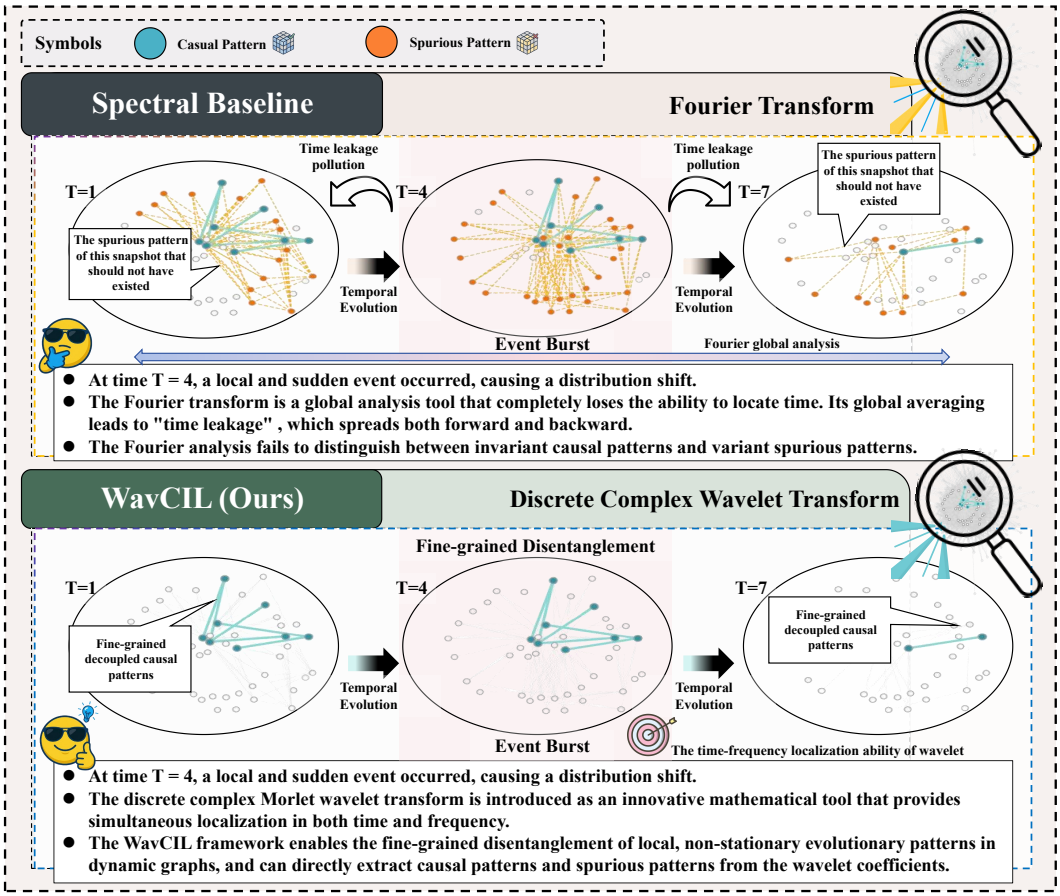

Figure 10: Case study of WavCIL's fine-grained disentanglement under real-world distribution shifts.

E.1 Limitations of Spectral-Domain Baseline Methods

As illustrated in the upper half of Figure 10, conventional spectral-domain invariant learning methods based on Fourier transform suffer from inherent limitations when handling such distribution shifts. The global integral property of the Fourier transform inherently prevents it from accurately localizing frequency components along the time axis. When a high-intensity distribution shift occurs at T_4, the averaging effect of global filtering induces noticeable temporal leakage. This phenomenon causes spurious patterns that are only coupled with specific contexts at a given time step to erroneously leak into both the historical (T_1) and the future (T_7) network snapshots. For example, a MOOC course may suddenly surge in popularity and draw a large number of students. This indicates that global

spectral analysis fails to achieve precise disentanglement in complex real-world topologies, and consequently cannot effectively characterize non-stationary distribution shifts driven by local events or phased changes, revealing a bottleneck when dealing with complex time-varying patterns[30].

E.2 Fine-Grained Disentanglement of WavCIL in the Wavelet Coefficient Domain

The lower half of Figure 10 presents the disentanglement results achieved by the WavCIL framework. By introducing the Discrete Complex Wavelet Transform (DCWT), WavCIL gains the ability to jointly localize in both time and frequency for the evolutionary signals of dynamic graphs. At the exact time step T_4 where the distribution shift occurs, WavCIL achieves fine-grained disentanglement of local and non-stationary evolutionary patterns in dynamic graphs. It selectively filters out spurious variant patterns coupled with specific contexts, while preserving cross-environment stable causal patterns(e.g., students' long-term consistent learning habits and interactions). More importantly, this local feature separation does not interfere with the topological structures at historical (T_1) or future (T_7) time steps. These results demonstrate that WavCIL achieves precise identification and generalizable utilization of local time-varying causal patterns in dynamic graphs, and validates the necessity of extending to the wavelet coefficient domain with time-frequency localization properties.

F Limitations

Our current work focuses solely on distribution shifts occurring in dynamic graphs, requiring explicit graph topology information and temporal node features. This inevitably restricts the direct application of WavCIL to purely sequential data, such as multivariate time series forecasting or scenarios missing graph structures. In the future, we plan to extend WavCIL with topology-agnostic adaptation strategies to handle non-stationary distribution shifts in broader sequential data.

G Related Works

Dynamic Graph Representation Learning Dynamic graphs are extensively exist in real-world scenarios and widely deployed in complex tasks such as event prediction and large-scale financial system analysis [38, 39]. Different from static graphs that primarily focus on fixed topological structures[40], dynamic graphs encapsulate abundant temporal evolutionary information [41, 42]. In recent years, considerable research efforts have been dedicated to developing Dynamic Graph Neural Networks (DyGNNs) for capturing the complex structural and feature dynamics of dynamic graphs [43]. Specifically, DySAT firstly introduces the self-attention mechanism to calculate dynamic aggregation weights of node features [44]; TGN integrates a memory module to continuously update node interaction states in continuous time [23]. Additionally, CAW innovatively leverages causal anonymous walks to extract inductive representations of temporal networks [45, 46]. Apart from the explorations in the conventional spatio-temporal domain, researchers have begun to focus on frequency-domain features. For example, spectral temporal graph neural networks have been effectively applied to multivariate time series forecasting [47], and global graph wavelet convolutions have been introduced to capture long-range global interactions in dynamic evolution [48, 22].

Out-of-Distribution Generalization on Graphs Most existing graph machine learning methods are established under the strict independent and identically distributed (i.i.d.) assumption that training and test data follow the same distribution [49, 50]. Nevertheless, uncontrollable environmental evolution in real-world scenarios tend to cause significant distribution shifts, causing traditional models to suffer severe performance degradation. [6, 51]. To address this challenge, abundant research progress has been achieved in out-of-distribution (OOD) generalization on static graphs. Existing studies first focus on invariant rationales, information bottlenecks, graph contrastive learning, and subgraph disagreement to learn stable graph representations under distribution shifts [52, 53, 54, 55]. Another line of work improves robustness through adversarial perturbation, distributionally robust optimization, and adaptive reweighting strategies [56, 57, 58, 59, 60]. More recent efforts further explore invariant information maximization, graph pre-training, and size generalization [36, 61, 62]. Prompt tuning, self-supervised disentanglement, and graph domain transfer are also adopted to enhance transferability across graph domains [63, 64, 65, 66, 67]. For instance, Shift-Robust GNN proposes to overcome distribution discrepancies in local graph training data by regularizing hidden layers [68]; OOD-GNN employs a nonlinear decorrelation mechanism to balance confounding factors

[69]; some studies further handle environment shifts on graph structures from the essence of the invariance perspective [70]. In the field of pure time series data, methods such as AdaRNN have been proposed to adaptively handle temporal distribution shifts in dynamic forecasting [71, 72].

Graph Invariant Learning In contrast, distribution shifts on dynamic graphs are more complex, typically involving a dual entanglement of spatial topology and non-stationary temporal evolution. Invariant learning provides a more solid theoretical foundation for addressing OOD challenges on dynamic graph structures [24, 73]. Its core lies in discovering and leveraging causal mechanisms that remain stable across environments[74, 75, 76, 77]—a principle exemplified by invariant risk minimization (IRM), which achieves this by optimizing predictive risk across distinct environments [78, 79]; related studies further reduce spurious correlations through debiasing, environment augmentation, and causal independence constraints[80, 81, 82, 83]. In the context of graph learning, identifying and extracting invariant subgraphs has emerged as one of the most effective strategies for eliminating spurious correlations [84, 85], while disentangled graph representation learning further separates independent latent factors [86, 87, 88, 89] and transferable structures [90, 91, 92]. DIDA pioneers a spatio-temporal intervention mechanism via decoupled spatio-temporal attention networks to effectively eliminate spurious correlations coupled with specific contexts [1]; EAGLE dynamically models latent environments through an environment-aware decoupling mechanism, thereby enhancing the generalization capability of models on unseen time periods [4]. The spatio-temporal domain methods above struggle to cope with more complex frequency-domain shifts, and spectral-domain invariant learning sheds new light on tackling this issue. SILD innovatively extends graph invariant learning to the spectral domain and conducts adaptive feature intervention by capturing invariant and variant spectral patterns within time-varying graph [5]. Moreover, frequency-domain analysis tools such as the Fourier transform have been widely adopted in time series and graph representation learning due to their powerful capability to model global evolution patterns [93, 94]. For instance, FedFormer successfully performs precise feature decomposition and forecasting of long-range time series in the frequency domain [95]. However, as indicated by relevant signal processing theories,the global Fourier transform loses temporal locality during frequency-domain analysis [96, 97]. This limitation makes it difficult for conventional spectral methods to accurately capture non-stationary shifts driven by transient or localized events [98]. Although multi-wavelets have been applied to learn differential operators in continuous-time systems [99],and Graph Wavelet Neural Networks (GWNN) have long been used for local information aggregation on static graphs [100], how to leverage time-frequency localized representations for OOD generalization on dynamic graphs remains a critical unsolved research gap [101, 102]. Our proposed WavCIL framework fills this gap. By introducing the discrete complex Morlet wavelet transform, we achieve fine-grained disentanglement of non-stationary causal patterns from spurious environmental patterns in the wavelet coefficient domain.

H Broader Impacts

This paper aims to advance the field of dynamic graph representation learning by proposing a OOD generalization framework. Our WavCIL inherently disentangles stable causal patterns from spurious correlations, offering strong capability against real-world distribution shifts driven by unpredictable time-varying factors such as emergency incidents and seasonal trends. This capability has positive societal implications by significantly enhancing the reliability, safety, and trustworthiness of DyGNNs in critical temporal systems, such as financial transaction risk forecasting and social network analysis. Furthermore, we anticipate that our work will not pose direct social or ethical negative impacts.

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: The abstract and introduction clearly and logically present the contributions and scope of our WavCIL.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: We discuss the limitations of WavCIL, focusing primarily on boundary effects for extremely short time series, their impact on disentanglement precision, and potential boundary compensation strategies in Appendix F.

   Guidelines:

   - The answer [N/A] 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: [Yes]

Justification: We provide detailed mathematical derivations and theoretical formulations in Appendix B, where the foundations of the discrete complex Morlet wavelet and invariant learning are rigorously established, thereby ensuring the validity and rigor of the analysis.

Guidelines:

- The answer [N/A] 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: We provide a complete implementation setup in Appendix D. Additionally, we provide the code for WavCIL in Appendix A.1 via a repository at `https://github.com/yantongzhu/WavCIL` and supplementary materials.

Guidelines:

- The answer [N/A] 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: We provide the code for WavCIL in Appendix A.1 via a repository at `https://github.com/yantongzhu/WavCIL` and supplementary materials.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide a complete implementation setup in Appendix D.

   Guidelines:

   - The answer [N/A] 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: In all tables, we report the standard deviation across multiple independent runs to indicate statistical significance when presenting and analyzing the experimental results.

   Guidelines:

   - The answer [N/A] 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: We provide the complete implementation setup including computer resources in Appendix D.

   Guidelines:

   - The answer [N/A] 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 and have ensured full compliance with its guidelines.

   Guidelines:

   - The answer [N/A] 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: The broader impacts are discussed in the Appendix H.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No such risks are associated with this work.

    Guidelines:

    - The answer [N/A] 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: All resources used in this work have been properly attributed to their original creators and rights owners.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: This work does not involve crowdsourcing or research involving human subjects.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: This paper is not related to this issue.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: This paper utilized LLMs exclusively for writing and formatting purposes. LLMs did not influence the core research methodology, scientific validity, or the originality of the contributions.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.