Graph-Budget Reduction as a Prerequisite for Scalable GNN Training on Decomposed Financial Time Series

Agustín M. de los Riscos[1,2⋆], Ana Lazcano[2], and Julio E. Sandubete[2]

[1] Faculty of Statistical Studies, Universidad Complutense de Madrid,
Avda. Complutense, s/n, 28040 Madrid, Spain
ariscos@ucm.es
[2] Faculty of Law, Business and Government, Universidad Francisco de Vitoria,
Carretera de Pozuelo a Majadahonda km. 1.800,
28223 Pozuelo de Alarcón, Madrid, Spain

Abstract. Graph neural networks (GNNs) can exploit the relational structure of financial markets, but only after raw signals are encoded as graphs whose elements can dominate memory and communication in distributed training clusters. We propose and evaluate a multiscale encoding *workflow* on financial assets. Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) separates each series into intrinsic mode functions (IMFs) and a residue; each component is mapped to natural visibility graphs (NVG), horizontal visibility graphs (HVG), and recurrence graphs; and the oscillatory IMF panel is optionally compressed through independent component analysis (ICA) before graph construction. Experiments characterize decomposition quality, reconstructive fidelity, and per-channel graph-budget statistics as a structural prerequisite for downstream component-wise GNN training, establishing a basis for future assessment of distributed execution, graph partitioning, and predictive performance on the resulting representations. Rank-k ICA halves the number of independent oscillatory graph channels and sharply reduces their aggregated edge counts on that block, while the monotonic residue—unchanged by ICA—is encoded in a separate channel, yielding complexity-controlled graph representations intended to support scalable financial time-series modeling with GNNs.

Keywords: Machine Learning · Graph Neural Networks · Empirical Mode Decomposition · Dimensionality Reduction

1 Introduction

Accurate modeling of financial time series underpins portfolio allocation, risk management, and systematic trading, yet market data rarely behave as stationary linear processes. Global equity indices, sector exchange-traded funds (ETFs), and safe-haven assets such as gold exhibit prolonged trends, regime shifts, volatility

⋆ Corresponding author

clustering, and scale-dependent oscillations that frustrate models trained directly on raw prices or returns [4,11]. Extracting predictive structure from such signals therefore requires representations that separate temporal scales adaptively and expose relational patterns amenable to modern deep learning.

Existing approaches only partially address this need. Spectral and wavelet methods rely on fixed basis functions that may misalign with regime-dependent financial dynamics. Classical Empirical Mode Decomposition (EMD) offers a data-driven alternative, but standard EMD suffers from mode mixing and sensitivity to noise; graph-based forecasting, in turn, typically operates on a single time-series encoding and seldom connects decomposition quality to downstream graph complexity [8,3]. Moreover, embedding each decomposed component as a separate graph on the full observation grid multiplies nodes and edges across modes. When each channel is processed independently in downstream modeling, the channel count sets the number of parallel graph workloads and thus the aggregate memory footprint across a cluster [14,9,7]. Prior work rarely integrates systematic benchmarking of EMD variants, comparative graph encodings per intrinsic mode function (IMF), and orthogonality-preserving compression prior to graph construction.

The proposed pipeline applies CEEMDAN to obtain oscillatory IMFs and a monotonic residue, optionally compresses the IMF panel with ICA, and maps each retained component to NVG, HVG, and recurrence graphs as a preprocessing stage for GNN modeling. **Scope.** The present study evaluates decomposition quality, reconstructive fidelity after ICA, and per-channel graph-budget diagnostics ($|V|$, $|E|$, channel count, connected components) as structural prerequisites for forthcoming component-wise GNN training, distributed execution assessment, and predictive benchmarking on the resulting representations. Downstream modeling is envisaged as *one isolated GNN per retained temporal component*: each graph is built over the full time index ($|V| \approx T$), preserving within-channel temporal integrity, with no cross-IMF or cross-score message passing. Parallelism across a distributed cluster is therefore obtained through independent component-wise workloads rather than by partitioning a single monolithic series graph; reducing channel count lowers the number of such parallel jobs and their aggregate edge budget. That structural link is argued here as a basis for future distributed training experiments. The main contributions are:

- Feasibility assessment of EMD on global indices, sector ETFs, and gold, validating residual behaviour and reconstruction quality.
- Benchmark of EMD, EEMD, and CEEMDAN on synthetic stress tests using mean cross-IMF coupling diagnostics ($\bar{\rho}$), with CEEMDAN selected on a composite multi-component scenario and confirmed qualitatively on empirical decompositions.
- Systematic multiscale graph transformation (NVG, HVG, recurrence) with topological diagnostics per IMF, separating oscillatory and residue budgets.
- ICA compression of the oscillatory IMF panel preserving reconstructive fidelity and score decorrelation, with explicit pre/post cardinality comparison.
- Graph-budget analysis quantifying per-channel workloads as a prerequisite for component-wise GNN training.

Section 2 reviews related work; Sections 3–5 present the framework, experiments, and conclusions.

2 Related Work

2.1 Empirical Mode Decomposition in Financial Time Series

The analysis of financial time series is inherently complex due to their non-stationary, nonlinear, and multiscale nature. Traditional Fourier-based methods often fail to capture the transient dynamics and volatility clustering characteristic of market data. To address these limitations, Huang et al. [4] introduced Empirical Mode Decomposition (EMD), a data-driven technique that adaptively decomposes signals into a set of intrinsic mode functions (IMFs). While foundational, the original EMD algorithm frequently suffers from mode mixing, where signals of different scales are collocated in a single IMF or a single scale is spread across multiple IMFs.

EMD seeks an additive representation

$$x(t) = \sum_{j=1}^{n} c_j(t) + r_n(t), \tag{1}$$

where $c_j(t)$ is the j-th IMF, $r_n(t)$ is the final residue, and n is the number of extracted oscillatory modes. Classical EMD iteratively subtracts envelope means until IMF conditions are met.

To mitigate mode mixing, Wu and Huang [12] proposed Ensemble EMD (EEMD), which adds white noise to the signal. Further refinement led to Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) by Torres et al. [11]. Let $\mathcal{E}_j\{\cdot\}$ return the j-th IMF, $\{\omega^{(i)}(t)\}_{i=1}^{I}$ be I noise realizations, and $\varepsilon_j > 0$ the noise scale at stage j. With $r_0(t) = x(t)$, each stage computes

$$\tilde{c}_j^{(i)}(t) = \begin{cases} \mathcal{E}_1\big\{r_0(t) + \varepsilon_0\, \mathcal{E}_1\{\omega^{(i)}(t)\}\big\}, & j = 1, \\ \mathcal{E}_j\big\{r_{j-1}(t) + \varepsilon_{j-1}\, \mathcal{E}_j\{\omega^{(i)}(t)\}\big\}, & j \geq 2, \end{cases} \tag{2}$$

The stage-j IMF is the trial average $c_j(t) = \frac{1}{I}\sum_{i=1}^{I} \tilde{c}_j^{(i)}(t)$, and the residue is updated as $r_j(t) = r_{j-1}(t) - c_j(t)$ until $r_j(t)$ is monotonic.

In financial contexts, CEEMDAN isolates short-term fluctuations from long-term trends, yet most pipelines use EMD outputs for denoising or sequence models without linking multiscale structure to graph-based architectures. Comparative EMD/EEMD/CEEMDAN benchmarks rarely combine residue diagnostics with cross-IMF coupling when selecting a decomposition backend.

2.2 Time Series to Graph Transformations

Mapping temporal sequences into the graph domain allows complex network theory to capture the underlying geometry of time series. Lacasa et al. [8]

pioneered the Natural Visibility Graph (NVG), which connects time points based on their relative visibility in the amplitude–time plane. For nodes $i < j$, the NVG adjacency is

$$A_{ij}^{\mathrm{NVG}} = \mathbb{1}\Big[y_k < y_i + \tfrac{t_k - t_i}{t_j - t_i}(y_j - y_i)\ \forall k \in \{i{+}1, \ldots, j{-}1\}\Big], \tag{3}$$

where $\mathbb{1}[\cdot]$ is the indicator. Luque et al. [10] introduced the Horizontal Visibility Graph (HVG),

$$A_{ij}^{\mathrm{HVG}} = \mathbb{1}[y_k < \min(y_i, y_j)\ \forall k \in \{i{+}1, \ldots, j{-}1\}], \tag{4}$$

a geometrically simpler variant that remains sensitive to the stochastic versus deterministic nature of the data while offering greater analytical tractability.

Beyond visibility-based methods, recurrence plots and their corresponding recurrence graphs, as described by Eckmann et al. [2], provide a phase-space perspective. With delay embedding $\mathbf{X}_i = [y_i, y_{i+\tau}, \ldots, y_{i+(d-1)\tau}]^\top$,

$$A_{ij}^{\mathrm{REC}} = \mathbb{1}[\|\mathbf{X}_i - \mathbf{X}_j\| \le \varepsilon], \tag{5}$$

where (τ, d, ε) are estimated per component via mutual information and false-nearest-neighbor criteria. Despite applications to regime detection, systematic comparisons of NVG, HVG, and recurrence encodings across individual IMFs remain scarce.

2.3 Graph Neural Networks for Financial Forecasting

Graph neural networks (GNNs) have emerged as a powerful paradigm for forecasting by exploiting relational dependencies among assets or temporal states. Gregnanin et al. [3] combined visibility graphs with dynamic GNNs and attention mechanisms to forecast stock prices, highlighting the ability of graph representations to capture nonlinear dependencies beyond conventional sequence models. Similarly, Bajracharya et al. [1] reported competitive performance from Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs) built on visibility-based transformations for local market indices.

Graph construction itself remains a critical challenge. Forecasting models such as DGDNN [13] learn dynamic graphs via signal-driven edge generation, but they address predictive accuracy rather than distributed training infrastructure. Systems such as DistDGL [14], PaGraph [9], and METIS-based graph partitioning [7] show that communication cost and partition quality depend on $|V|$, $|E|$, and graph connectivity—quantities that multiply when each decomposed component is encoded as an independent graph channel. The integration of multiscale decomposition with GNNs is beginning to gain traction; for instance, Zheng and Zhufu [15] proposed a cross-scale GNN that combines frequency-domain sequence attention with graph message passing. Nevertheless, the structural cost of large-scale graph construction and the node–edge growth that arises when embedding multiple decomposed scales remain largely unaddressed in current financial forecasting frameworks.

2.4 Dimensionality Reduction and Distributed GNN Considerations

When downstream modeling treats each retained component as an independent graph workload, the channel count—and thus the aggregate $|V|$, $|E|$, and memory footprint across parallel jobs—scales with the number of IMFs or ICA scores. Principal Component Analysis (PCA) is often employed to reduce input redundancy, yet for multichannel IMF panels Independent Component Analysis (ICA), and specifically the FastICA algorithm of Hyvärinen and Oja [5], offers a complementary alternative by enforcing statistical independence rather than mere decorrelation. For a standardized oscillatory panel $\mathbf{X} \in \mathbb{R}^{T\times n}$,

$$\mathbf{z}_t = \mathbf{W}\,\tilde{\mathbf{x}}_t, \qquad \hat{\mathbf{X}} = \mathbf{Z}\mathbf{B}^\top, \tag{6}$$

where $\mathbf{z}_t \in \mathbb{R}^k$ are independent-component scores at rank $k < n$, $\mathbf{W}$ is the unmixing matrix, and $\hat{\mathbf{X}}$ approximates $\mathbf{X}$. At a fixed rank, PCA and ICA can yield numerically identical reconstructions when both reduce to the same orthogonal subspace; ICA is nevertheless preferred here because the retained scores are explicitly decorrelated and interpretable as independent sources prior to graph encoding.

Recent work indicates that interaction complexity must be managed for scalability [6], yet ICA-based IMF compression before graph construction—and per-channel graph-budget control prior to component-wise GNN training—remains under-investigated.

2.5 Synthesis of Research Gaps

In summary, while the components for a multiscale graph-based pipeline exist independently, their systematic integration—with joint EMD benchmarking, per-IMF graph diagnostics, and ICA-based budget control prior to GNN training—remains incomplete in the financial forecasting literature.

3 Proposed Framework

3.1 Overview

Figure 1 summarizes the proposed multiscale encoding workflow. Processing begins with a financial time series $x(t)$, which CEEMDAN decomposes into IMFs and a residue (Eqs. (1)–(2) in Section 2), thereby separating oscillatory dynamics across scales from the slow trend. The oscillatory IMF panel $\mathbf{X} \in \mathbb{R}^{T\times n}$ (residue excluded, columns standardized) may then be compressed through optional FastICA at rank $k < n$ (Eq. (6)), reducing the number of graph channels while limiting cross-mode redundancy. Each retained temporal component—whether an IMF, the residue, or an ICA-derived score Z_j—is subsequently encoded as an NVG, an HVG, and a recurrence graph (Eqs. (3)–(5)), yielding three independently processable graph instances per channel. Reported $|V|$, $|E|$, and channel counts quantify the cost of these independent encoding workloads; they

are not measured partition quality or communication volume. The contribution is the *integration order* and the empirical characterization of per-channel graph budgets across scales and encoders, not new decomposition or graph-construction formulas.

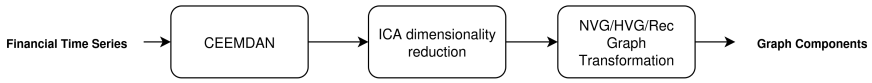


Fig. 1. Proposed multiscale pipeline: CEEMDAN decomposition, optional ICA compression of the oscillatory IMF panel, and triple graph encoding (NVG, HVG, recurrence) per retained component.

3.2 Pipeline stages

Decomposition. CEEMDAN is applied to daily closing prices with ensemble size I=100, noise scale ε=0.05 (relative to signal standard deviation), cap `max_imf`=14, and a fixed random seed. Decomposition stops when the residue is monotonic (PyEMD implementation). Section 4 benchmarks CEEMDAN against EMD and EEMD on synthetic signals before applying it to empirical assets.

Optional compression. FastICA with the `logcosh` nonlinearity, unit-variance whitening, `max_iter=3000`, and a fixed random seed projects the standardized oscillatory block to k scores. Reconstruction fidelity is measured by per-IMF RMSE and R^2 after mapping scores back to native IMF units. At k=4 on MSCI World, PCA yields numerically identical reconstructions but different score trajectories; ICA is retained for interpretable independent sources.

Graph encoding. For each component series $\{y_i\}_{i=1}^{T}$, three graphs $G = (V, E)$ are built with $V = \{1, \ldots, T\}$ and node feature y_i. Recurrence parameters (τ, d, ε) are estimated *per component* (`random_state=42`): τ from the first local minimum of mutual information ($\tau_{\max}$=50); d from the false-nearest-neighbour criterion ($d_{\max}$=10, FNN ratio $< 10\%$); ε as the 10th percentile of pairwise Euclidean embedding distances. Each graph is summarized by $|V|$, $|E|$, edge density, connected-component count, and average clustering coefficient (Table 2). ICA at rank k reduces oscillatory channels from n to k; the residue is always retained separately.

4 Experiments and Results

4.1 Data and EMD applicability

Experiments use daily closing prices (2012–2026) for MSCI World, three sector ETFs (XLE, XLP, XLV), and spot gold (XAUUSD). EMD prerequisites—non-stationary prices with stationary returns, nonlinearity, and volatility clustering—are verified empirically for all series before decomposition.

Table 1. Synthetic benchmark: N = number of IMFs before the residue; $\bar{\rho}$ = mean absolute cross-IMF Pearson correlation (stress-test row in bold).

Family	EMD N	EMD $\bar{\rho}$	EEMD N	EEMD $\bar{\rho}$	CEEMDAN N	CEEMDAN $\bar{\rho}$
Close tones	6	0.099	12	0.072	8	0.090
Linear chirp	4	0.080	12	0.089	7	0.071
Burst on carrier	4	0.071	11	0.103	8	0.108
Close tones + chirp + noise	8	0.064	12	0.092	8	0.087
Multi-component stress test	8	0.040	11	0.071	8	**0.038**

4.2 Synthetic benchmark and model selection (CEEMDAN)

To compare EMD variants in a controlled environment, a synthetic benchmark is introduced. Multiple signal families are generated to stress known difficulties in decomposition—closely spaced frequencies, chirp-like non-stationarity, intermittent bursts on a carrier, additive noise, and an aggregated multi-component stress test with an equity-like slow drift. Synthetic signals have length T=2500; all methods share `max_imf`=12.

For each family, EMD, EEMD, and CEEMDAN are compared using the mean absolute Pearson correlation across IMF pairs ($\bar{\rho}$). In Table 1, N denotes the number of IMFs returned *before* the residue. EEMD often reaches the N=12 cap because input noise creates additional extrema; CEEMDAN typically returns $N \in \{7, 8\}$ under the same cap.

Model selection. The decomposition variant is chosen from Table 1 using only the *multi-component stress test* row. That composite signal superposes the difficulties exercised separately in the preceding families—closely spaced tones, chirp-like non-stationarity, intermittent bursts, and additive noise—on an equity-like slow drift, and therefore offers a single benchmark that jointly captures the behaviours most relevant to the financial series analysed below. On this scenario CEEMDAN attains the lowest $\bar{\rho}$ among the three variants (0.038 vs. EMD 0.040 and EEMD 0.071); the gap to EMD is small on a single deterministic run and is not treated as decisive evidence of lower mode mixing in isolation. The remaining rows are retained as diagnostics: EMD can attain lower $\bar{\rho}$ on isolated families (e.g., burst on carrier), but each stresses one mechanism in isolation and is not used for selection. The same CEEMDAN configuration is then applied to all empirical assets; as shown in the next subsection (Figure 2), CEEMDAN also allocates oscillatory structure more cleanly across IMFs on real prices and yields a smoother, monotone residue with less high-frequency roughness than EEMD, providing the qualitative confirmation that motivates retaining CEEMDAN despite the near-tied $\bar{\rho}$ on the synthetic stress test.

4.3 Decomposition outcomes and dimensionality reduction

On all empirical assets, CEEMDAN provides near-exact numerical reconstruction when IMFs and residue are summed. Relative reconstruction errors are below 4.1×10^{-5} on MSCI and below 10^{-16} on the ETFs and gold. All series use the same

closing-price representation; the twelve-order-of-magnitude gap reflects floating-point tolerance relative to price magnitude (index level $\sim 10^2$) rather than a different input convention. On MSCI, CEEMDAN yields a monotone, trend-like residual (Figure 2), consistent with the cleaner trend–oscillation separation sought in the synthetic benchmark.

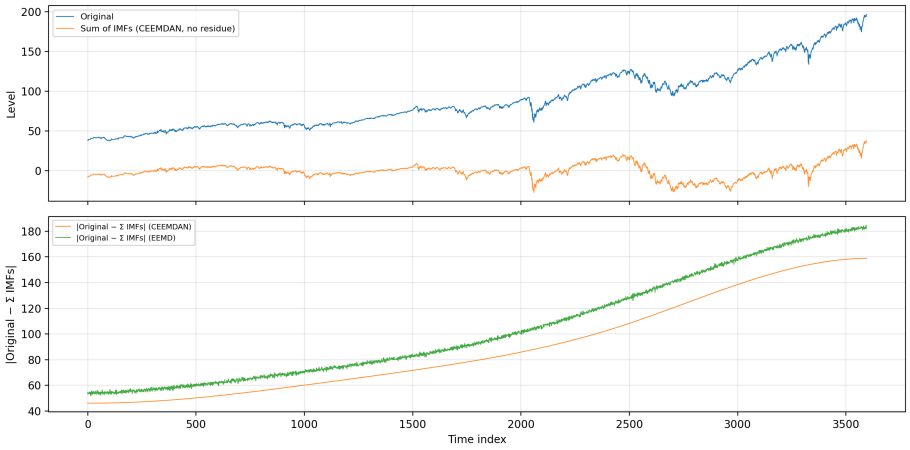


Fig. 2. CEEMDAN decomposition of MSCI World: reconstruction using the sum of IMFs (excluding the residue) and absolute reconstruction gap, compared against EEMD. CEEMDAN yields a monotone residue consistent with a long-term trend, whereas EEMD retains higher-frequency variability in the residue.

On MSCI World, the oscillatory CEEMDAN block comprises eight IMFs. FastICA ranks $k \in \{3, 4, 5\}$ are evaluated on this panel using mean R^2 and Frobenius relative error after mapping scores back to native IMF units; off-diagonal score correlation remains below 10^{-14} throughout. At k=3, mean R^2=0.508 and Frobenius error 0.662; at k=4, 0.636/0.589; at k=5, 0.759/0.567. Rank k=4 is selected as the preferred operating point because it offers the best balance between reconstructive fidelity and compression: it halves the number of oscillatory channels (8→4, −50%), substantially improving fidelity over k=3, while retaining stronger compression than k=5 (8→5, −37.5%) at the cost of only a moderate increase in reconstruction error. The additional graph channel required at k=5—and its associated triple-encoding workload—further reinforces this choice. Figure 3 visualizes the IMF-level reconstruction gap at k=4; Table 3 reports the paired pre/post ICA graph budgets on the same oscillatory block.

With rank k=4 fixed, each retained oscillatory component is prepared for triple graph encoding (Section 3.2). Table 2 reports the recurrence parameters (τ, d, ε) estimated independently for every native IMF and for each corresponding ICA score Z_j. On native IMFs, τ ranges from 2 (IMF_2) to 34 (IMF_7) and d decreases at coarser scales (4 down to 2). After compression, all four scores share

Table 2. MSCI World recurrence parameters (τ, d, ε) estimated independently per component (Section 3.2); oscillatory block only (residue encoded separately).

Native CEEMDAN IMFs

Comp.	τ	d	ε
IMF_1	9	4	0.185
IMF_2	2	4	0.156
IMF_3	3	4	0.135
IMF_4	7	4	0.189
IMF_5	15	4	0.256
IMF_6	29	3	0.189
IMF_7	34	2	0.080
IMF_8	32	2	0.122

FastICA k=4 scores

Comp.	τ	d	ε
Z_1	28	4	0.308
Z_2	7	4	0.305
Z_3	11	4	0.197
Z_4	3	4	0.259

Table 3. MSCI World oscillatory-block graph budgets before ICA (native CEEMDAN IMFs) and after ICA (k=4 scores); triple encoding (NVG, HVG, recurrence) per channel. Residue excluded.

Block	Channels	$\sum \lvert E \rvert$ HVG	$\sum \lvert E \rvert$ NVG	$\sum \lvert E \rvert$ Rec.
Oscillatory (native IMFs)	8	55,414	6,366,824	137,965
Oscillatory (ICA Z_j)	4	28,645	57,449	135,346

d=4 but retain distinct (τ, ε) pairs, confirming that ICA alters the delay structure of individual channels rather than merely relabelling existing IMFs.

Table 3 compares oscillatory-block graph budgets *before* ICA (eight native IMFs) and *after* compression (four FastICA scores at k=4) on MSCI World under triple encoding (NVG, HVG, and recurrence per channel). Channel count falls by half (8→4, −50%), halving the number of parallel oscillatory workloads. Aggregated HVG edges decrease from 55,414 to 28,645 (−48%); aggregated NVG edges from 6.37×10^6 to 5.74×10^4 (−99%), reflecting both fewer channels and the change in local extrema structure induced by ICA mixing. Aggregated recurrence edges remain essentially unchanged (137,965 vs. 135,346), so the main cardinality gain on this block comes from visibility encoders rather than from recurrence. Together with mean R^2=0.636 at k=4, this documents the reconstruction–cardinality trade-off at the preprocessing stage. The monotonic residue is encoded separately and is excluded from this comparison.

Design alternatives. A raw-price single-encoder baseline—one NVG, HVG, or recurrence graph on the undecomposed series—would minimize channel count but sacrifice explicit scale separation and the intended parallel multiscale architecture. Triple encoding is retained to expose cross-encoder structural contrasts at the cost of 3× multiplicity per channel. At k=4 on MSCI, PCA matches FastICA in reconstruction error, but ICA is preferred for decorrelated score trajectories aligned with independent downstream channels. CEEMDAN is chosen over EMD/EEMD on the composite synthetic stress test (Section 4.2) before ICA is applied. Fuller ablations—single- vs. multi-encoder pipelines and workflows with vs. without CEEMDAN—are left to future work where page limits preclude additional experiments.

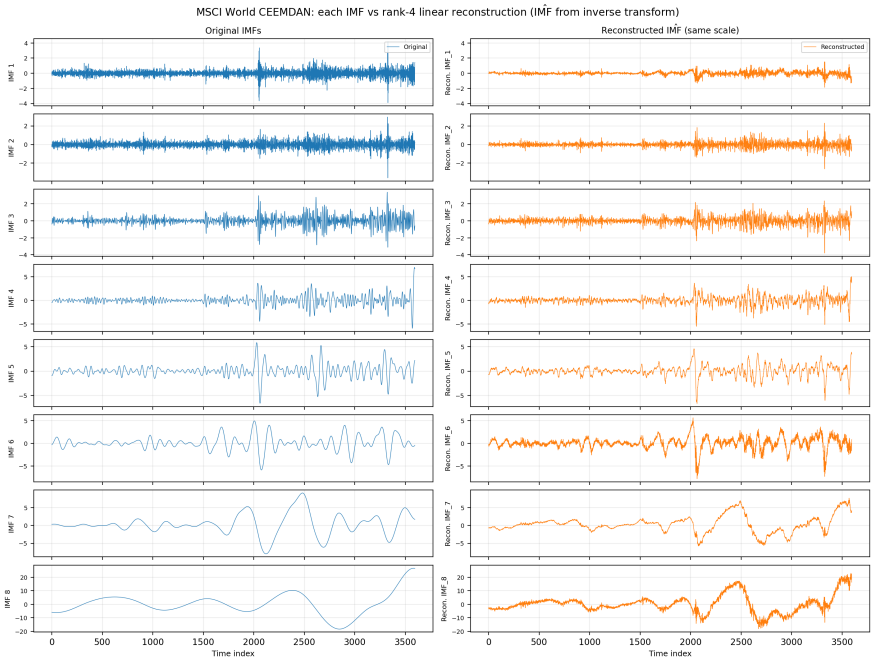


Fig. 3. MSCI World (oscillatory block): original IMFs (left) vs. rank-k=4 linear reconstructions (right) with *shared y-axis per IMF row*. The visible gap reflects information loss before graph encoding.

4.4 Time-series to graph transformations

The downstream design envisages *independent* processing of each retained component: every IMF, ICA score, or residue is mapped to its own NVG, HVG, and recurrence graphs over the full observation grid (Table 2), with no cross-channel message passing at the modeling stage—one isolated GNN per component channel and encoder view. The diagnostics below therefore characterize per-channel structural budgets—$|V|$, $|E|$, connected-component counts, and clustering—not fused multi-scale training cost or measured distributed-partition savings. ICA reduces the number of such parallel workloads on the oscillatory block (Table 3). Intra-graph distributed partitioning [14] would apply only as an exceptional fallback if a single component graph exceeded device memory; the primary scalability lever in the intended architecture is fewer independent full-length component graphs executed in parallel.

Oscillatory IMFs on MSCI ($|V|\approx$3587 per channel): HVG graphs remain sparse and connected ($|E| \in [6266, 7161]$). NVG edge counts grow from 10,695 (IMF_1) to 1,162,388 (IMF_8). Recurrence graphs maintain $|E| \sim 10^4$ but fragment into hundreds to thousands of connected components (e.g., 1590 on IMF_3); this reflects encoder topology within a single channel, not distributed partition savings, although each component can be treated as a separate subgraph if needed.

Residue (reported separately, outside oscillatory budgets): on MSCI the residue NVG reaches $|E| \approx 4.9 \times 10^6$, dominating any single oscillatory channel and clarifying why Table 3 reports oscillatory budgets only. The residue is encoded with NVG, HVG, and recurrence graphs but is unaffected by ICA compression of the oscillatory block.

Average clustering decreases across HVG IMFs (0.658→0.389) and varies across NVG/recurrence encoders, with recurrence fragmentation differing from visibility graphs.

4.5 Reproducibility and limitations

On empirical assets, CEEMDAN uses ensemble size I=100, noise scale ε=0.05 (relative to the signal standard deviation), and `max_imf`=14, stopping when the residue is monotonic; on the synthetic benchmark (Section 4.2), I=180, ε=0.03, and `max_imf`=12. FastICA applies the `logcosh` nonlinearity, unit-variance whitening, and `max_iter`=3000. Both stages use a fixed random seed for reproducibility. All figures use a single seeded run; multi-seed variance will be assessed in future work. The scope is a representation and preprocessing workflow whose reported budgets characterize independent per-channel graph workloads as inputs to forthcoming component-wise GNN training, predictive benchmarking, and distributed execution studies. Code, input data, and pipeline scripts are available at `https://github.com/MultiEMDGraphsTS/paper-multiscale-emd-graphs`.

5 Conclusions and Future Work

This study presents a multiscale encoding *workflow*—a representation and preprocessing pipeline that establishes diagnostic evidence for scalable GNN modeling—combining CEEMDAN, optional FastICA compression, and triple graph encoding for financial time series. Experiments characterize decomposition quality, reconstructive fidelity after ICA, and per-channel graph-budget statistics on global indices, sector ETFs, and gold as prerequisites for future component-wise GNN training and predictive benchmarking. At k=4, FastICA halves independent oscillatory channels (8→4) while preserving mean R^2=0.636; on MSCI, aggregated oscillatory HVG and NVG edges fall by 48% and 99%, respectively (Table 3), whereas recurrence totals change little. The residue remains a separate channel unaffected by ICA. The envisaged downstream stage assigns one isolated GNN per retained component and encoder view on full-length graphs, without cross-channel message passing, so oscillatory-block budget reduction lowers the number of parallel workloads. Future work includes component-wise GNN training and predictive benchmarking, optional late fusion of channel-level predictions, multi-seed robustness analysis, ablations against single-encoder and non-decomposed baselines, assessment of distributed execution and graph partitioning, and intra-graph partitioning only where a single component graph exceeds device memory.

References

1. Bajracharya, S., Dahal, N., Tuladhar, Y., et al.: Ai-powered stock forecasting: A graph-based approach for NEPSE. Journal of Computing **1**(1-2), 1–18 (2025)
2. Eckmann, J.P., Kamphorst, S.O., Ruelle, D.: Recurrence plots of dynamical systems. Europhysics Letters **4**(9), 973 (1987)
3. Gregnanin, M., et al.: Stock price time series forecasting using dynamic graph neural networks and attention mechanism in recurrent neural networks. In: International Conference on Artificial Neural Networks. pp. 313–328. Springer (2024)
4. Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Quanan, Z., Yen, N.C., Chi, C.C., Liu, W.H.: The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences **454**(1971), 903–995 (1998)
5. Hyvärinen, A., Oja, E.: Independent component analysis: algorithms and applications. Neural Networks **13**(4-5), 411–430 (2000)
6. Jia, S., Gao, H., Huang, J., et al.: CIRGNN: Leveraging cross-chart relationships with a graph neural network for stock price prediction. Mathematics **13**(15), 2402 (2025). https://doi.org/10.3390/math13152402
7. Karypis, G., Kumar, V.: METIS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Tech. Rep. 95-035, University of Minnesota, Department of Computer Science (1998)
8. Lacasa, L., Luque, B., Ballesteros, F., Luque, J., Nuñez, A.: From time series to complex networks: The visibility graph. Proceedings of the National Academy of Sciences **105**(13), 4972–4975 (2008)
9. Lin, Z., Li, C., Miao, Y., Liu, Y., Xu, Y.: PaGraph: Scaling GNN training on large graphs via computation-aware caching. In: Proceedings of the 11th ACM Symposium on Cloud Computing (SoCC). pp. 401–415 (2020). https://doi.org/10.1145/3419111.3421281
10. Luque, B., Lacasa, L., Ballesteros, F., Luque, J.: Horizontal visibility graphs: Exact results for random time series. Physical Review E **80**(4), 046103 (2009)
11. Torres, M.E., Colominas, M.A., Schlotthauer, G., Flandrin, P.: A complete ensemble empirical mode decomposition with adaptive noise. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4144–4147. IEEE (2011)
12. Wu, Z., Huang, N.E.: Ensemble empirical mode decomposition: a noise-assisted data analysis method. Advances in Adaptive Data Analysis **1**(01), 1–41 (2009)
13. You, Z., Shi, Z., Bo, H., et al.: DGDNN: Decoupled graph diffusion neural network for stock movement prediction. arXiv preprint arXiv:2401.01846 (2024)
14. Zheng, D., Song, X., Ma, C., Tan, Z., Ye, Z., Dong, J.S., Xiong, H., Zhang, Z., Karypis, G.: DistDGL: Distributed graph neural network training for billion-scale graphs. In: 2020 IEEE/ACM Workshop on Irregular Applications: Architectures and Algorithms (IA3). pp. 36–44. IEEE (2020)
15. Zheng, K., Zhufu, H.: A stock price forecasting model based on frequency-domain sequence attention and cross-scale graph neural networks. In: 2025 International Conference on Data Engineering and Communication Systems (DECS). pp. 1145–1155. IEEE (2025)