FreDN: Spectral Disentanglement for Time Series Forecasting via Learnable Frequency Decomposition

Zhongde An[1], Jinhong You[1], Jiyanglin Li[2], Yiming Tang[3], Wen Li[4*], Heming Du[5*], Shouguo Du[6]

[1]Shanghai University of Finance and Economics, Shanghai, China
[2]Guizhou University of Finance and Economics, Guiyang, China
[3]Shanghai Lixin University of Accounting and Finance, Shanghai, China
[4]Shanghai University of International Business and Economics, Shanghai, China
[5]Australian National University, Canberra, Australia
[6]Shanghai Municipal Big Data Center, Shanghai, China
{2023213372@stu.sufe.edu.cn, johnyou07@163.com, jylli@mail.gufe.edu.cn, jstangyiming@163.com, wen.li.sh@outlook.com, heming.du@anu.edu.au, shouguo.du.sh@outlook.com}

Abstract

Time series forecasting is essential in a wide range of real world applications. Recently, frequency-domain methods have attracted increasing interest for their ability to capture global dependencies. However, when applied to non-stationary time series, these methods encounter the *spectral entanglement* and the computational burden of complex-valued learning. The *spectral entanglement* refers to the overlap of trends, periodicities, and noise across the spectrum due to *spectral leakage* and the presence of non-stationarity. However, existing decompositions are not suited to resolving spectral entanglement. To address this, we propose the Frequency Decomposition Network (FreDN), which introduces a learnable Frequency Disentangler module to separate trend and periodic components directly in the frequency domain. Furthermore, we propose a theoretically supported ReIm Block to reduce the complexity of complex-valued operations while maintaining performance. We also re-examine the frequency-domain loss function and provide new theoretical insights into its effectiveness. Extensive experiments on seven long-term forecasting benchmarks demonstrate that FreDN outperforms state-of-the-art methods by up to 10%. Furthermore, compared with standard complex-valued architectures, our real-imaginary shared-parameter design reduces the parameter count and computational cost by at least 50%.

Introduction

Time series forecasting is essential in energy, finance, and traffic management. Time-domain forecasting models continue to make notable progress in recent years. MLP-based architectures like TimeMixer (Wang et al. 2024) and SOFTS (Han et al. 2024) leverage multiscale mixing or centralized channel fusion to improve efficiency and scalability. At the same time, frequency-domain methods have gained traction for their ability to capture global dependencies and suppress noise. Models like FreTS (Yi et al. 2023) suggest that the frequency spectrum offers MLPs a more complete view of the signal, facilitating the learning of global patterns. Recent works such as FITS (Xu, Zeng, and Xu 2024) and FreDF (Wang et al. 2025a) further demonstrate that frequency-based representations enhance forecasting accuracy.

Frequency-domain techniques have become increasingly prevalent in time series forecasting, with the Fourier transform (Oppenheim 1999) playing a central role in analyzing spectral structures. However, when applied to non-stationary real world time series, frequency-domain methods face two major difficulties: *spectral entanglement* that hampers effective decomposition, and the computational burden of learning with complex-valued representations. *Spectral entanglement* is the overlap of trend, periodic, and noise components in the frequency domain, as illustrated in Figure 1 (d). This issue arises due to two main factors. First, the finite look-back window length L introduces *spectral leakage*: the resolution of the Discrete Fourier Transform (DFT) is limited by L according to the Nyquist theorem (Oppenheim 1999). Any true frequency that does not align with the discrete basis causes energy to spread across all frequencies. Second, non-stationary trends, which do not exhibit periodicity, cannot be accurately represented by a small set of frequencies, as illustrated in Figure 1 (b).

The *spectral entanglement* undermines the effectiveness of conventional trend-seasonality decomposition methods (Cleveland et al. 1990), which are typically employed to address non-stationarity in time series. For example, the moving average (Wu et al. 2021; Wang et al. 2024) can be viewed as applying a fixed low-pass filter in the frequency domain, but this approach is frequency-index-based and fails to resolve spectral entanglement, as shown in Figure 1 (e). Frequency-domain approaches such as Top-K selection (Wu et al. 2021; Zhou et al. 2022b) retain only dominant frequencies by amplitude, often discarding leaked or overlapping components (Figure 1 (f)). Moreover, frequency-domain learning involves complex-valued coefficients. Prior works such as FEDformer (Zhou et al. 2022b) and FreTS (Yi et al. 2023) apply specialized matrix operations to complex numbers, but these increase model complexity and hinder the integration of standard real-valued neural networks.

*Corresponding authors

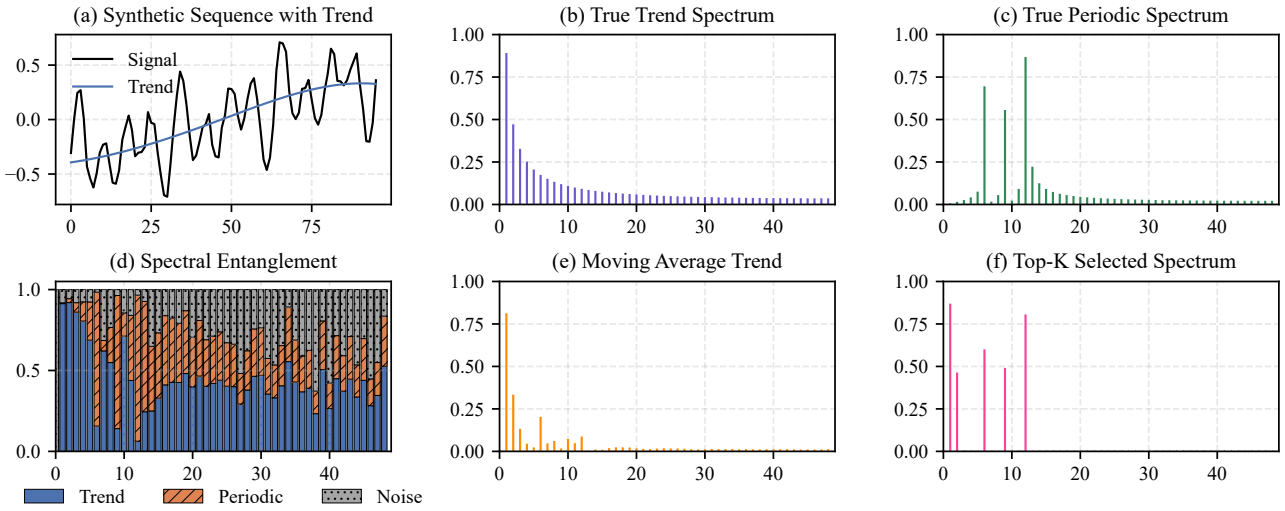

Figure 1: Spectral component entanglement. Top row: (a) the synthetic sequence with trend generated by B-spline; (b) the frequency spectrum of the real trend; (c) the frequency spectrum of the real periodicity. Bottom row: (d) the frequency-wise proportion of trend, periodic, and noise in the original synthetic sequence; (e) the spectrum of the trend extracted by the moving average in the time domain; (f) the spectrum of the TopK-selected frequency components.

In response to the challenges mentioned earlier, we introduce Frequency Decomposition Network (FreDN), a novel approach that integrates learnable Frequency Disentangler and ReIm Block. At its core, FreDN uses a **Frequency Disentangler** that separates trend and seasonal components across the entire frequency spectrum, reducing spectral overlap. The disentangled trend components are processed in the time domain, where gradual variations are better captured. Seasonal components, are modeled in the frequency domain using a theory-guided **ReIm Block**, which avoids direct complex-valued computation by decomposing complex representations into real and imaginary parts with shared weights, reducing parameter cost by over 50% without degrading performance. We further conduct a theoretical analysis comparing the gradient properties of loss functions defined in the time and frequency domains, highlighting the structured and informative nature of frequency-domain MAE. The main contributions of this work are as follows:

- We highlight spectral entanglement as an overlooked issue in frequency-domain learning and proposes a simple yet effective solution to mitigate its impact through a *Frequency Disentangler*.
- We propose a theory-guided *ReIm Block* that models real and imaginary parts as dual real-valued branches. This design reduces parameter count by over 50% without sacrificing performance, and offers a practical pathway to adapt real-valued architectures for learning in the complex plane.
- We present a theoretical analysis comparing the gradient propagation behaviors of MSE and MAE losses in both time and frequency domains, highlighting the structured gradients induced by frequency-domain MAE.

Related Work

Time Series Forecasting

Time series forecasting plays a crucial role in fields such as finance, energy, and climate science. Classical statistical models like ARMA, ARIMA, and VAR (Box et al. 2015; Makridakis and Hibon 1997; Watson 1994; Brockwell 1991) rely on linear assumptions and are limited in capturing nonlinear dependencies. Deep learning methods significantly improve modeling capacity, with RNNs (Salinas et al. 2020), CNNs (Bai, Kolter, and Koltun 2018; Liu et al. 2022), Transformers (Wu et al. 2021; Zhou et al. 2021; Nie et al. 2023; Liu et al. 2024; Wen et al. 2023), and other task-specific architectures (Du, Yu, and Zheng 2020, 2021; Du, Du, and Li 2023; Du et al. 2023) demonstrating strong performance on complex temporal patterns. Recent efforts emphasize efficiency and fairness through compact designs (Wang et al. 2024; Han et al. 2024; Lin et al. 2024; Qiu et al. 2024, 2025b,a). Frequency-domain modeling has gained attention for its ability to capture global periodic structures and suppress noise (Zhang et al. 2025). Approaches like FreTS, FITS, and FreDF (Yi et al. 2023; Xu, Zeng, and Xu 2024; Wang et al. 2025a) operate directly on spectral representations, while models such as CoST and FiLM (Woo et al. 2022; Zhou et al. 2022a) incorporate frequency-domain operations into neural architectures. Broader spectral paradigms, such as Fourier Neural Operators and wavelet CNNs (Li et al. 2021; Fang et al. 2024), demonstrate strong modeling capacity.

Time Series Decomposition

To handle non-stationarity, time-domain methods often decompose sequences into trend, seasonal, and noise components. Classical approaches include differencing (Oppenheim 1999), STL (Cleveland et al. 1990), SSA (Golyand-

ina, Nekrutkin, and Zhigljavsky 2001), and EMD (Huang et al. 1998). While effective, these techniques are tailored to time-domain processing and do not adapt well to frequency-domain tasks. Non-stationary sequences pose additional challenges in the frequency domain, where overlapping spectral energy from trend and periodic components causes *spectral entanglement* (Zhang et al. 2025). Heuristic solutions like Top-K frequency selection (Wu et al. 2021; Zhou et al. 2022b) are limited by spectral leakage and lack structural interpretability. Though localized spectral methods like STFT (Oppenheim 1999) offer partial solutions, they are constrained by fixed resolution or high computational cost. Unlike our FreDN, which targets spectral entanglement via trend-seasonal separation, recent methods like FreDo (Sun and Boning 2022), FAITH (Li et al. 2025), and FreqMoE (Liu 2025) either model frequencies holistically or decompose signals differently without resolving frequency overlap. To address these limitations, we propose FreDN, a learnable frequency-domain decomposition framework that disentangles trend and periodic components via adaptive frequency disentangler, while preserving compatibility with real-valued networks.

Method

Preliminary

Problem Definition. The input multivariate time series is $X \in \mathbb{R}^{C\times L}$, where C is the number of features and L is the look-back window length. The target of forecasting is $Y \in \mathbb{R}^{C\times\tau}$, where τ denotes the prediction horizon. Our forecasting model is denoted as $\hat{Y} = f_\theta(X)$, where f_θ represents the parameterized forecasting model. Moreover, $L_{\text{freq}} = \lfloor \frac{L}{2} \rfloor + 1$ and $\tau_{\text{freq}} = \lfloor \frac{\tau}{2} \rfloor + 1$ denote the frequency-domain lengths of the input and target sequences after real-valued FFT. We denote the real and imaginary parts of any complex-valued variable $\tilde{Z}$ by $\tilde{Z}_r$ and $\tilde{Z}_i$.

The overall architecture of the FreDN model is illustrated in Figure 2.

Reversible Instance Normalization. We normalize the input using Reversible Instance Normalization (RevIN) (Kim et al. 2021), following the common practice in many models (Nie et al. 2023; Liu et al. 2024; Han et al. 2024).

Series Embedding. To increase the model's expressiveness, the input $X \in \mathbb{R}^{C\times L}$ is reshaped to $X \in \mathbb{R}^{C\times L\times 1}$ and projected into a d-dimensional embedding space: $X_{\text{emb}} = X \odot \phi_d \in \mathbb{R}^{C\times L\times d}$ with learnable ϕ_d.

Discrete Fourier Transformation. Given an input $X \in \mathbb{R}^L$, the DFT is defined as:

$$\tilde{X}_k = \sum_{t=0}^{L-1} X_t \cdot e^{-j2\pi kt/L}, \quad k = 0, 1, \ldots, L-1,$$

where j is the imaginary unit. The DFT is a linear transformation and can be expressed as: $\tilde{X} = F \cdot X$, where $F \in \mathbb{C}^{L\times L}$ is the Fourier matrix satisfies the unitary condition $F \cdot F^H = I$, where F^H is the conjugate transpose of F. We use the Fast Fourier Transform (FFT) as an efficient algorithm for computing the DFT.

Frequency Disentangler

Due to the spectral entanglement across all frequencies, existing methods such as moving average and Top-K selection (Zhou et al. 2022b; Wu et al. 2021) fail to function effectively in the frequency domain. To address this issue, we design a learnable Frequency Disentangler that adaptively separates the entire spectrum into smooth trend and seasonal components, effectively alleviating spectral entanglement.

The following theorem shows that smooth trends have non-zero spectral energy at all frequencies and decay gradually with frequency index k. Intuitively, smooth trend functions cannot be accurately represented by a finite set of periodic Fourier bases; their spectra typically consist of energy-concentrated low frequencies and small-magnitude high frequencies. Detailed proofs are provided in Appendix D.

Theorem 1 (Spectral of Sobolev-Smooth Trends)**.** *Let $f \in W^{m,2}([0,1])$ be a sobolev-smooth function with square-integrable m-th derivative for some $m \geq 1$. $\hat{f}(k)$ is the k-th Fourier coefficient of the periodic extension of f. Then there exists a constant $C > 0$ such that for all $k \in \mathbb{Z} \setminus \{0\}$,*

$$0 \leq |\hat{f}(k)| \leq \frac{C}{|k|^m}.$$

This sobolev-smooth assumption encompasses a wide class of commonly used trend representations, including B-splines, local polynomial regression, and kernel-based smoothers, all of which produce functions within Sobolev spaces $W^{m,2}$ for moderate m. The resulting trend component is sufficiently smooth without being overly restrictive.

Moreover, due to the finite window length used in DFT, *spectral leakage* inevitably occurs, spreading the energy of seasonal component across adjacent frequencies. This leads to entanglement of trend and seasonality across the entire spectrum. To address this entanglement, we adopt a learnable decomposition strategy that assigns trend and seasonal attribution to each frequency component independently. This design allows the model to flexibly adapt to the intrinsic spectral structure of the input.

Given the embedded input $X_{\text{emb}} \in \mathbb{R}^{C\times L\times d}$, we apply FFT along the temporal axis and obtain:

$$\tilde{X} = \text{FFT}(X_{\text{emb}}) \in \mathbb{C}^{C\times L_{\text{freq}}\times d}.$$

We then introduce a learnable frequency disentangler $M \in \mathbb{R}^{L_{\text{freq}}\times d}$, and define:

$$\tilde{X}_{\text{trend}} = \tilde{X} \odot \sigma(M), \quad \tilde{X}_{\text{season}} = \tilde{X} \odot (1 - \sigma(M)),$$

where σ is the sigmoid function and $\odot$ denotes element-wise multiplication. The trend component is transformed back via inverse FFT, while the seasonal part $\tilde{X}_{\text{season}}$ remains in the frequency domain for further modeling. The decay pattern of sobolev-smooth trend also guides the initialization of the disentangler. For instance, when $m = 1$, one may initialize the trend weights using $w(k) = -\log(1 + |k|)$, which approximates the expected spectral decay of smooth trends.

Our learnable Frequency Disentangler performs separation of trend and seasonality at each frequency, achieving fine-grained disentanglement of spectral components and

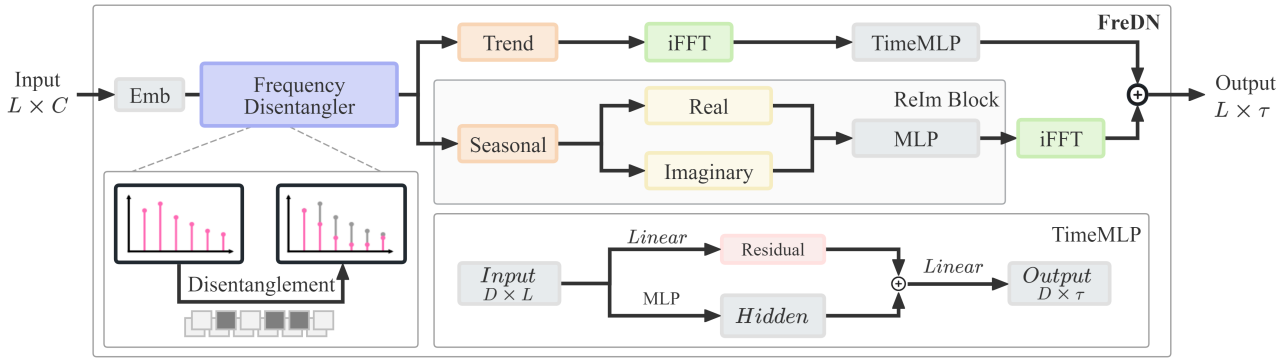

Figure 2: Overview of the proposed FreDN architecture. The input is first embedded by adding a redundant dimension, then decomposed by a learnable Frequency Disentangler into trend and seasonal components. The trend is transformed back to the time domain and processed by a TimeMLP. The seasonal real and imaginary parts are separately modeled by a shared MLP under the ReIm Block. The final prediction sums both outputs.

yielding a smoothly varying trend spectrum as shown in Figure 1 (a) (b) alongside a decomposed, relatively stationary seasonal spectrum.

TimeMLP. We adopt a standard MLP with residual connection, where the input passes through a multilayer perceptron and a linear residual path before being projected to the target length. Specifically, TimeMLP learns smooth trend representations in the time domain. Detailed architectures are provided in Appendix B.

ReIm Block

While trend patterns are naturally modeled in the time domain, periodic characteristics are more effectively captured in the complex plane. Existing methods such as FreTS (Yi et al. 2023) and FITS (Xu, Zeng, and Xu 2024) commonly adopt standard complex-valued linear layers to process frequency components. However, this approach requires specially designed complex matrix multiplications, introducing additional implementation complexity and computational overhead.

A standard complex-valued linear layer takes the form:

$$\tilde{Y} = (W_r + jW_i)(\tilde{X}_r + j\tilde{X}_i),$$

where W_r and W_i are real-valued. To understand the role of each component, we consider the following result.

Definition 1 (Complex Linear Projection)**.** *Let $\tilde{X} = \tilde{X}_r + j\tilde{X}_i \in \mathbb{C}^d$ be a complex-valued input, and let $W = W_r + jW_i \in \mathbb{C}^{1\times d}$ be a complex weight vector. Then the output $\tilde{Y} = W\tilde{X}$ can be written as:*

$$\tilde{Y} = W_r\tilde{X} + jW_i\tilde{X}.$$

Here, $W_r\tilde{X}$ is a real-weighted projection of the input, and $jW_i\tilde{X}$ performs an orthogonal rotation 90° of the same input with separate weights.

This implies that the standard complex projection linearly combines the original input and its quadrature component, allowing for arbitrary complex rotations. The imaginary unit j provides a canonical mechanism for representing a phase rotation of 90°.

If we set $W_i = 0$, the operation degenerates to $\tilde{Y} = W_r\tilde{X}$, which corresponds to the projection strategy used in the proposed ReIm Block. Instead of relying on full complex-valued multiplications, ReIm Block adopts shared real-valued weights, significantly simplifying implementation while retaining sufficient expressiveness. The following theorem formalizes its representational capacity:

Theorem 2 (Expressiveness of ReIm Block)**.** *Let $\tilde{X} = [\tilde{X}_1, \ldots, \tilde{X}_d]^\top \in \mathbb{C}^d$ be a complex-valued input vector.*

- ***(Complex Linear Projection)** Let $W \in \mathbb{C}^d$. Then the standard complex projection $\tilde{Y}_{complex} = W^\top\tilde{X}$ can represent any complex value.*
- ***(ReIm Block)** Let $W_r \in \mathbb{R}^d$, the projection$\tilde{Y}_{real} = W_r^\top\tilde{X}$ can represent any complex number in $\mathbb{C}$ if and only if there exist at least two entries $\tilde{X}_i, \tilde{X}_j$ such that their phase difference satisfies $\arg(\tilde{X}_i) - \arg(\tilde{X}_j) \notin \pi\mathbb{Z}$. That is, the input contains at least two linearly independent directions in the complex plane.*

Intuitively, the complex plane is a two-dimensional space, so any complex number can be represented as a linear combination of two orthogonal complex numbers (i.e., with a 90° phase difference).

Given the theoretical justification above, we define the ReIm Block as:

$$\tilde{Y}_{\text{season}} = \text{MLP}(\tilde{X}^{(r)}_{\text{season}}) + j \cdot \text{MLP}(\tilde{X}^{(i)}_{\text{season}}),$$

Due to the spectral complexity of real-world time series and the increasing number of frequency components with longer inputs, a real-valued projection as in ReIm Block is sufficient to approximate a wide range of complex representations through linear combinations.

Finally, we obtain the prediction by inverse transforming and combining with the trend:

$$\hat{Y} = \text{TimeMLP}(X_{\text{trend}}) + \text{IFFT}(\tilde{Y}_{\text{season}}).$$

Loss Function Analysis

Here, we provide a gradient-based analysis to examine why the frequency-domain MAE loss, first introduced in

Models	FreDN (Ours)		FreDF (2025)		SOFTS (2024)		iTransformer (2024)		TimeMixer (2023)		FreTS (2023)		PatchTST (2023)		DLinear (2023)	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.397**	**0.420**	0.482	0.473	0.421	0.436	0.438	0.448	0.430	0.440	0.463	0.463	0.424	0.440	0.416	0.435
ETTh2	**0.330**	**0.380**	0.371	0.405	0.361	0.401	0.377	0.406	0.359	0.399	0.462	0.467	0.351	0.391	0.483	0.472
ETTm1	**0.340**	**0.372**	0.375	0.395	0.360	0.389	0.366	0.393	0.362	0.386	0.370	0.391	0.363	0.384	0.355	0.378
ETTm2	**0.243**	**0.304**	0.259	0.318	0.273	0.325	0.270	0.329	0.260	0.317	0.274	0.333	0.254	0.316	0.255	0.316
Weather	**0.215**	**0.253**	0.233	0.270	0.242	0.276	0.237	0.272	0.228	0.266	0.224	0.274	0.235	0.274	0.238	0.289
Electricity	**0.155**	**0.247**	0.160	0.253	0.161	0.256	0.161	0.257	0.165	0.258	0.166	0.300	0.164	0.283	0.164	0.297
Traffic	0.381	0.261	0.411	0.290	**0.376**	**0.260**	0.379	0.270	0.386	0.267	0.431	0.300	0.402	0.283	0.423	0.297
1st Count	6	6	0	0	1	1	0	0	0	0	0	0	0	0	0	0
2nd Count	0	1	1	1	0	0	1	0	0	1	1	0	2	2	2	3

Table 1: Multivariate forecasting results averaged over prediction horizons $\tau \in \{96, 192, 336, 720\}$. For each τ, the best result is selected from lookback lengths $L \in \{96, 192, 336, 512, 720\}$ for all models. The best results are in bold, and the second best are underlined.

Model	Metric	ETT	Weather	Electricity	Traffic
FreDN	MSE	**0.328**	**0.215**	**0.155**	**0.381**
	MAE	**0.369**	**0.253**	**0.247**	**0.261**
MovDN	MSE	0.337	0.219	0.158	0.388
	MAE	0.373	0.254	0.250	0.269
TopKDN	MSE	0.338	0.222	**0.155**	0.389
	MAE	0.374	0.258	0.248	0.268

Table 2: Comparison of decomposition methods. Results are averaged over prediction lengths $\tau \in \{96, 192, 336, 720\}$, with ETT representing the average across ETTh1, ETTh2, ETTm1, and ETTm2.

FreDF (Wang et al. 2025a), may offer more efficient optimization behavior.

$$\mathcal{L}_F = \frac{1}{\tau_{\text{freq}}} \|\tilde{\epsilon}\|_1.$$

FreDF (Wang et al. 2025a) claims that "by transforming the label sequence into this orthogonal frequency domain, the dependence from label autocorrelation could be effectively mitigated". This statement is statistically inaccurate. As a linear transformation, DFT preserves the correlation structure of residuals, unless the residuals are strictly independent. This misunderstands the role of orthogonality: orthogonality enables frequency separation, not decorrelation.

Despite this, the frequency-domain MAE loss is highly effective in practice, particularly when integrated into spectral learning frameworks. We now examine how different loss functions affect learning by comparing their gradient behavior with respect to $\hat{Y}$:

Time-domain MSE:

$$\mathcal{L}_{\text{time-MSE}} = \frac{1}{\tau} \|\epsilon\|_2^2, \quad \frac{\partial \mathcal{L}_{\text{time-MSE}}}{\partial \hat{Y}} = \frac{2}{\tau} \cdot \epsilon$$

Frequency-domain MSE:

$$\mathcal{L}_{\text{freq-MSE}} = \frac{1}{\tau_{\text{freq}}} \|\tilde{\epsilon}\|_2^2, \quad \frac{\partial \mathcal{L}_{\text{freq-MSE}}}{\partial \hat{Y}} = \frac{2}{\tau_{\text{freq}}} \cdot \epsilon$$

Structure	Metric	ETT	Weather	Electricity	Traffic
Dual Complex	MSE	**0.328**	**0.215**	**0.155**	**0.381**
	MAE	**0.369**	**0.253**	**0.247**	**0.261**
Complex Linear	MSE	0.335	0.216	0.157	0.394
	MAE	0.371	**0.253**	0.250	0.270

Table 3: Performance comparison between Dual-Complex and Complex-Linear. ETT (avg) denotes the average of ETTh1, ETTh2, ETTm1, and ETTm2. Results are averaged over prediction lengths $\tau \in \{96, 192, 336, 720\}$.

Specifically, from the Parseval's theorem (Oppenheim 1999), the energy of the residual is preserved under DFT, which means $\|\epsilon\|_2^2 = \|\tilde{\epsilon}\|_2^2$. This implies that time-MSE and frequency-MSE differ only by a constant factor in both loss magnitude and gradient.

Time-domain MAE:

$$\mathcal{L}_{\text{time-MAE}} = \frac{1}{\tau} \|\epsilon\|_1, \quad \frac{\partial \mathcal{L}_{\text{time-MAE}}}{\partial \hat{Y}} = \frac{1}{\tau} \cdot \text{sign}(\epsilon)$$

Frequency-domain MAE:

$$\mathcal{L}_{\text{freq-MAE}} = \frac{1}{\tau_{\text{freq}}} \|\tilde{\epsilon}\|_1, \quad \frac{\partial \mathcal{L}_{\text{freq-MAE}}}{\partial \hat{Y}} = \frac{1}{\tau_{\text{freq}}} \cdot F^H \left(\frac{\tilde{\epsilon}}{|\tilde{\epsilon}|} \right)$$

In time-domain MAE, the gradient at each time step is a real number of unit magnitude, determined solely by the sign of the residual. This results in sparse, local signals, where each step receives an isolated update.

In contrast, frequency-domain MAE produces complex gradients of unit magnitude. The complex gradients carries a specific phase, which encodes the temporal shift (i.e., starting position) of a sinusoidal basis in the time domain, which modulates a global periodic component across the entire sequence. *As a result, frequency-domain MAE introduces structured, long-range interactions into the gradient signal, enabling the model to capture global temporal patterns aligned with dominant frequencies.*

Detailed proofs are provided in Appendix D.

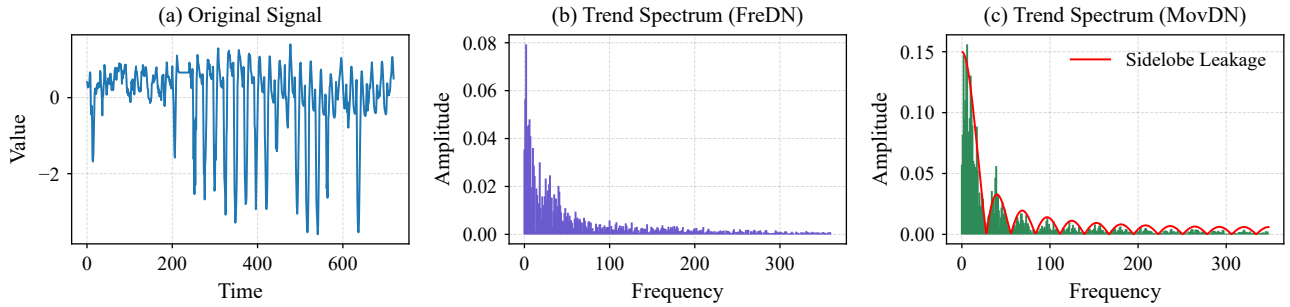

Figure 3: (a) Original signal from the ETTh1 with $L = 720$; (b) Trend spectrum in our FreDN; (c) Trend spectrum learned by MovDN using moving average, the red line indicating the theoretical sidelobe leakage introduced by the moving average filter.

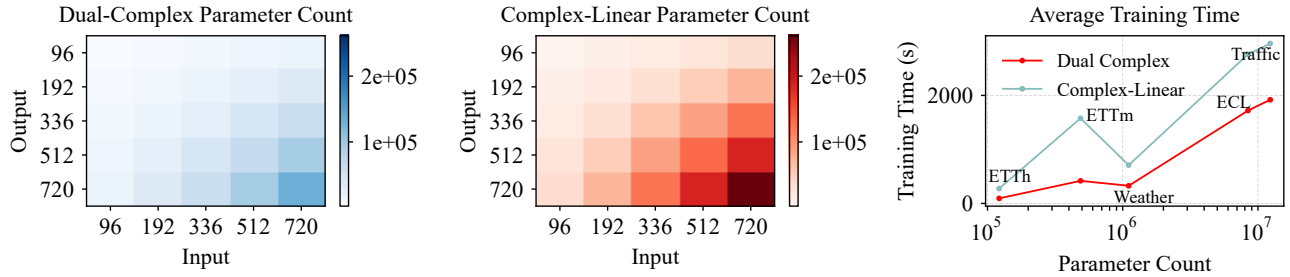

Figure 4: Efficiency of ReIm Block. (a) Parameter count of the ReIm Block under varying input/output lengths; (b) Parameter count of the Complex-Linear structure; (c) Average training time on different datasets.

Experiments

Datasets. To evaluate the performance of FreDN, we conduct extensive experiments on seven widely used long-term time series forecasting benchmarks: ETTh1, ETTh2, ETTm1, ETTm2, Weather, Electricity, and Traffic (Wu et al. 2021; Wang et al. 2025a; Zhou et al. 2021). Detailed dataset descriptions are provided in Appendix A.

Baselines. We compare our model against a set of representative Linear-based or MLP-based baselines, including SOFTS (Han et al. 2024), TimeMixer (Wang et al. 2024), DLinear (Zeng et al. 2023). Two Frequency-based methods include FreTS (Yi et al. 2023), FreDF (Wang et al. 2025a). We also consider Transformer-based methods including iTransformer (Liu et al. 2024) and PatchTST (Nie et al. 2023).

Implementation. The baseline models are reproduced using the official training scripts provided by FreDF (Wang et al. 2025a). All models are optimized using the Adam optimizer (Kingma and Ba 2015). For FreDN, we use only the frequency-domain MAE loss without the time and frequency domains fused loss structure employed by FreDF (Wang et al. 2025a). *To ensure fairness, we follow the common practice of selecting hyperparameters based on the prediction length. For all models, the same set of hyperparameters is applied across all input lengths* $\{96, 192, 336, 512, 720\}$ *for a given prediction length.* All experiments are implemented in PyTorch (Paszke et al. 2019) and conducted on a single NVIDIA GeForce RTX 3090 GPU with 24GB memory. More experimental details are provided in Appendix B.

Main results

Table 1 reports long-term forecasting results on seven benchmarks averaged over $\tau \in \{96, 192, 336, 720\}$. Each model uses default settings, and for every prediction length, the best result is selected across input lengths $L \in \{96, 192, 336, 512, 720\}$. FreDN achieves the best performance on most datasets. On average MSE, it surpasses FreDF (Wang et al. 2025a) and FreTS (Yi et al. 2023) by 10% and 14%, and outperforms recent time-domain MLPs SOFTS (Han et al. 2024) and TimeMixer (Wang et al. 2024) by 6%. Although our best results with lookback window selecting outperform TimeMixer++ (Wang et al. 2025b) (as shown in the Appendix C), we exclude it for fairness, since TimeMixer++ (Wang et al. 2025b) adopts a fixed $L = 96$ and its official implementation is not publicly available. The hyperparameter sensitivity analysis are provided in Appendix C. We also report the standard deviation of FreDN performance under five runs with different random seeds in Appendix E, which exhibits that the performance is stable. We further provide the visualization analysis in Appendix F.

Frequency Disentanglement Analysis

The comparison of different decomposition methods is shown in Table 2. FreDN adopts our proposed Frequency Disentangler. MovDN applies a moving average with a fixed window size of 25, which follows the strategy used in TimeMixer (Wang et al. 2024). TopKDN selects the frequency components with the largest K amplitudes as the periodic part. Detailed configurations of MovDN and TopKDN are provided in Appendix B.

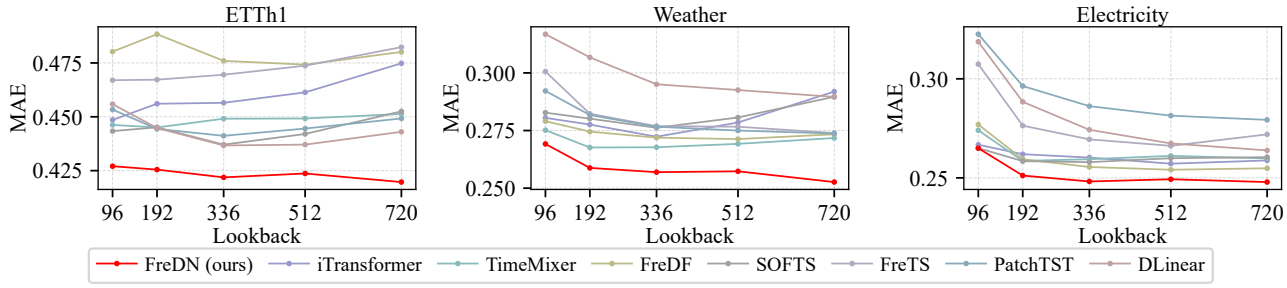

Figure 5: Effect of lookback window length on average MAE across all prediction horizons. FreDN maintains consistently superior accuracy under different lookback settings, with performance improving monotonically as the input length increases.

In Figure 3, we present the decomposition of real world sequence. Figure 3 (b) shows the trend spectrum obtained using the Frequency Disentangler. The trend obtained by moving average in Figure 3(c) exhibits significant *sidelobe leakage*, characterized by a dominant main lobe accompanied by multiple high-energy sidelobes. *This structure does not reflect the true spectral property of the trend component, but rather arises as a systematic artifact introduced by the decomposition method.* This is because the finite moving window causes spectral leakage in the frequency domain, which is a direct consequence of the Nyquist theorem (Oppenheim 1999). Specifically, the moving average decomposition is equivalent to a filter with the following frequency response:

$$H(f) = \frac{1}{k} \cdot \frac{\sin(\pi f k)}{\sin(\pi f)} \cdot e^{-j\pi f(k-1)}$$

where k represents the window length of the moving average, and $\frac{\sin(\pi f k)}{\sin(\pi f)}$ describes the low-pass filter characteristics with sidelobes, illustrated by the red curve in Figure 3 (c). This phenomenon further underscores the inadequacy of moving average techniques for frequency-based processing and supports the necessity of our Frequency Disentangler to decompose directly in the frequency domain.

Efficiency of ReIm Block

In Section **ReIm Block**, we provided a theoretical justification for the ReIm Block. To further demonstrate its practical efficiency, we compare the parameter count and average training time of the proposed ReIm Block with a Complex-Linear structure. The Complex-Linear structure replaces the linear projections in MLP with complex-valued linears, where the detailed architecture is described in Appendix B. Specifically, to implement a Complex-Linear, it requires two sets of parameters for real and imaginary parts, and follows the complex matrix rule (Golub and Van Loan 2013):

$$\tilde{W} \cdot \tilde{X} = \tilde{W}^{(r)}\tilde{X}^{(r)} - \tilde{W}^{(i)}\tilde{X}^{(i)} + j \cdot (\tilde{W}^{(r)}\tilde{X}^{(i)} + \tilde{W}^{(i)}\tilde{X}^{(r)}).$$

As shown in Figure 4, FreDN with ReIm Block reduces the parameter count by approximately 50% under the same configuration. Furthermore, it achieves significant training speedup, reducing average runtime by up to 75% on the ETTm dataset. Full results are provided in Appendix C.

Comparison of different complex-valued structures. We compare the average forecasting performance of the proposed ReIm Block with Complex-Linear structure in Table 3. Despite achieving a significant reduction in model size and training cost as demonstrated in Figure 4, the performance of the ReIm Block is not compromised, and even slightly outperforms the Complex-Linear structure. This may be attributed to the structural limitations of complex-valued representations, such as the lack of activation functions or layer normalization in complex form, highlighting the structural advantage of the ReIm Block design.

Effect of lookback window length. Although longer input sequences contain richer temporal information, lots of models tend to suffer from performance degradation as the lookback length increases. This is likely due to the weaker correlation between distant inputs and future targets, which poses challenges to the model's ability to capture global dependencies. We explore how varying the lookback length $L \in \{96, 192, 336, 512, 720\}$ affects forecasting accuracy across all datasets. Figure 5 presents the results, which shows FreDN achieving stable performance improvements as L increases. The performance improvement of FreDN over other models becomes more pronounced with longer inputs, indicating that FreDN is better equipped to exploit the additional information provided by extended lookback windows.

Conclusion

We present a novel frequency-domain perspective for non-stationary time series forecasting, focusing on spectral entanglement issue. To address this, we propose **FreDN**, which leverages a learnable Frequency Disentangler to disentangle trends and seasonalities directly in the frequency domain. We also design the ReIm Block block with a single real-valued MLP to model complex spectra efficiently. We theoretically justify the feasibility of ReIm Block and analyze the frequency-domain MAE loss through a gradient propagation view, showing its advantage in capturing global dependencies. ReIm Block provides an insightful perspective for frequency-domain modeling, which may inspire future architectural innovations. Empirically, FreDN ranks first in 26 out of 28 MAE tasks across four prediction lengths and seven baselines (see Appendix C).

References

Bai, S.; Kolter, J. Z.; and Koltun, V. 2018. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. arXiv:1803.01271.

Box, G. E.; Jenkins, G. M.; Reinsel, G. C.; and Ljung, G. M. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.

Brockwell, P. J. 1991. *Time series: Theory and methods*. Springer-Verlag.

Cleveland, R. B.; Cleveland, W. S.; McRae, J. E.; Terpenning, I.; et al. 1990. STL: A seasonal-trend decomposition. *J. off. Stat*, 6(1): 3–73.

Du, H.; Du, S.; and Li, W. 2023. Probabilistic time series forecasting with deep non-linear state space models. *CAAI Transactions on Intelligence Technology*, 8(1): 3–13.

Du, H.; Li, L.; Huang, Z.; and Yu, X. 2023. Object-Goal Visual Navigation via Effective Exploration of Relations Among Historical Navigation States. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2563–2573.

Du, H.; Yu, X.; and Zheng, L. 2020. Learning Object Relation Graph and Tentative Policy for Visual Navigation. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 19–34. Cham: Springer International Publishing. ISBN 978-3-030-58571-6.

Du, H.; Yu, X.; and Zheng, L. 2021. VTNet: Visual Transformer Network for Object Goal Navigation. arXiv:2105.09447.

Fang, Y.; Wang, Z.; Zhang, L.; Cao, J.; Chen, H.; and Xu, R. 2024. Spiking wavelet transformer. In *European conference on computer vision*, 19–37. Springer.

Golub, G. H.; and Van Loan, C. F. 2013. *Matrix computations*. JHU press.

Golyandina, N.; Nekrutkin, V.; and Zhigljavsky, A. 2001. *Analysis of Time Series Structure: SSA and Related Techniques*. Chapman and Hall/CRC.

Han, L.; Chen, X.-Y.; Ye, H.-J.; and Zhan, D.-C. 2024. SOFTS: Efficient Multivariate Time Series Forecasting with Series-Core Fusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Huang, N. E.; Shen, Z.; Long, S. R.; Wu, M. C.; Shih, H. H.; Zheng, Q.; Yen, N.-C.; Tung, C. C.; and Liu, H. H. 1998. 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.

Kim, T.; Kim, J.; Tae, Y.; Park, C.; Choi, J.-H.; and Choo, J. 2021. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *International conference on machine learning*.

Li, R.; Jiang, M.; Liu, Q.; Wang, K.; Feng, K.; Sun, Y.; and Zhou, X. 2025. FAITH: Frequency-domain Attention In Two Horizons for time series forecasting. *Knowledge-Based Systems*, 309: 112790.

Li, Z.; Kovachki, N.; Azizzadenesheli, K.; Liu, B.; Bhattacharya, K.; Stuart, A.; and Anandkumar, A. 2021. Fourier Neural Operator for Parametric Partial Differential Equations. arXiv:2010.08895.

Lin, S.; Lin, W.; Wu, W.; Chen, H.; and Yang, J. 2024. SparseTSF: Modeling Long-term Time Series Forecasting with *1k* Parameters. In *Forty-first International Conference on Machine Learning*.

Liu, M.; Zeng, A.; Chen, M.; Xu, Z.; Lai, Q.; Ma, L.; and Xu, Q. 2022. Scinet: Time series modeling and forecasting with sample convolution and interaction. *Advances in Neural Information Processing Systems*, 35: 5816–5828.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *The Twelfth International Conference on Learning Representations*.

Liu, Z. 2025. FreqMoE: Enhancing Time Series Forecasting through Frequency Decomposition Mixture of Experts. arXiv:2501.15125.

Makridakis, S.; and Hibon, M. 1997. ARMA models and the Box–Jenkins methodology. *Journal of forecasting*, 16(3): 147–163.

Nie, Y.; H. Nguyen, N.; Sinthong, P.; and Kalagnanam, J. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *International Conference on Learning Representations*.

Oppenheim, A. V. 1999. *Discrete-time signal processing*. Pearson Education India.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Qiu, X.; Hu, J.; Zhou, L.; Wu, X.; Du, J.; Zhang, B.; Guo, C.; Zhou, A.; Jensen, C. S.; Sheng, Z.; and Yang, B. 2024. TFB: Towards Comprehensive and Fair Benchmarking of Time Series Forecasting Methods. In *Proc. VLDB Endow.*, 2363–2377.

Qiu, X.; Wu, X.; Cheng, H.; Liu, X.; Guo, C.; Hu, J.; and Yang, B. 2025a. DBLoss: Decomposition-based Loss Function for Time Series Forecasting. In *NeurIPS*.

Qiu, X.; Wu, X.; Lin, Y.; Guo, C.; Hu, J.; and Yang, B. 2025b. DUET: Dual Clustering Enhanced Multivariate Time Series Forecasting. In *SIGKDD*, 1185–1196.

Salinas, D.; Flunkert, V.; Gasthaus, J.; and Januschowski, T. 2020. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191.

Sun, F.-K.; and Boning, D. S. 2022. FreDo: Frequency Domain-based Long-Term Time Series Forecasting. arXiv:2205.12301.

Wang, H.; Pan, L.; Shen, Y.; Chen, Z.; Yang, D.; Yang, Y.; Zhang, S.; Liu, X.; Li, H.; and Tao, D. 2025a. FreDF: Learning to Forecast in the Frequency Domain. In *The Thirteenth International Conference on Learning Representations*.

Wang, S.; LI, J.; Shi, X.; Ye, Z.; Mo, B.; Lin, W.; Shengtong, J.; Chu, Z.; and Jin, M. 2025b. TimeMixer++: A General Time Series Pattern Machine for Universal Predictive Analysis. In *The Thirteenth International Conference on Learning Representations*.

Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J. Y.; and ZHOU, J. 2024. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In *The Twelfth International Conference on Learning Representations*.

Watson, M. W. 1994. Vector autoregressions and cointegration. *Handbook of econometrics*, 4: 2843–2915.

Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; and Sun, L. 2023. Transformers in time series: a survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, 6778–6786.

Woo, G.; Liu, C.; Sahoo, D.; Kumar, A.; and Hoi, S. 2022. CoST: Contrastive Learning of Disentangled Seasonal-Trend Representations for Time Series Forecasting. In *International Conference on Learning Representations*.

Wu, H.; Xu, J.; Wang, J.; and Long, M. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34: 22419–22430.

Xu, Z.; Zeng, A.; and Xu, Q. 2024. FITS: Modeling Time Series with \$10k\$ Parameters. In *The Twelfth International Conference on Learning Representations*.

Yi, K.; Zhang, Q.; Fan, W.; Wang, S.; Wang, P.; He, H.; An, N.; Lian, D.; Cao, L.; and Niu, Z. 2023. Frequency-domain mlps are more effective learners in time series forecasting. *Advances in Neural Information Processing Systems*, 36: 76656–76679.

Zeng, A.; Chen, M.; Zhang, L.; and Xu, Q. 2023. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, 11121–11128.

Zhang, Q.; Sun, Y.; Wen, H.; Yang, P.; Li, X.; Li, M.; Lam, K.-Y.; Yiu, S.-M.; and Yin, H. 2025. Time Series Analysis in Frequency Domain: A Survey of Open Challenges, Opportunities and Benchmarks. arXiv:2504.07099.

Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 11106–11115.

Zhou, T.; Ma, Z.; Wang, X.; Wen, Q.; Sun, L.; Yao, T.; Yin, W.; and Jin, R. 2022a. FiLM: Frequency improved Legendre Memory Model for Long-term Time Series Forecasting. In *NeurIPS*.

Zhou, T.; Ma, Z.; Wen, Q.; Wang, X.; Sun, L.; and Jin, R. 2022b. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning*, 27268–27286. PMLR.