NJS-KAN: Neural Jacobi-Spectral KANs for Long-term Time Series Forecasting

1st Yucheng Wang
University of Science and Technology of China
Institute of Plasma Physics, Chinese Academy of Sciences
Hefei, Anhui, China
yucheng.wang@ipp.ac.cn

2nd Wenhui Hu*
Institute of Plasma Physics, Chinese Academy of Sciences
Hefei, Anhui, China
huwh@ipp.ac.cn
*Corresponding author

3rd Qiping Yuan
Institute of Plasma Physics, Chinese Academy of Sciences
University of Science and Technology of China
Hefei, Anhui, China
qpyuan@ipp.ac.cn

4th Bingjia Xiao
Institute of Plasma Physics, Chinese Academy of Sciences
University of Science and Technology of China
Hefei, Anhui, China
bjxiao@ipp.ac.cn

***Abstract*—Long-term Time Series Forecasting (LTSF) remains a challenging endeavor due to the intricate coupling of non-stationary dynamics and high-dimensional dependencies. While recent linear models prioritize efficiency, they often lack the capacity to model highly non-linear volatility, whereas standard Kolmogorov-Arnold Networks (KANs) face prohibitive parameter explosion and numerical instability when applied to multivariate tasks. To bridge this gap, we propose the Neural Jacobi-Spectral Kolmogorov-Arnold Network (NJS-KAN), a novel architecture designed to reconcile high-capacity non-linear modeling with computational efficiency. Our approach introduces a Low-Rank Jacobi-KAN layer that utilizes orthogonal Jacobi basis functions and imposes a CP-tensor decomposition on the weight space, effectively reducing parameter complexity from quadratic to linear while ensuring gradient stability. Furthermore, we design a Dual-Domain Gating mechanism that synergizes local temporal convolutions with global frequency-domain spectral features to overcome the spectral bias of local operators. Extensive experiments on six real-world benchmarks demonstrate that NJS-KAN achieves state-of-the-art performance, consistently outperforming recent Transformer and linear baselines in both predictive accuracy and memory efficiency. Code is available at https://github.com/hyfqphy/NJS-KAN.git.**

***Index Terms*—Long-term Time Series Forecasting, Kolmogorov-Arnold Networks, Low-Rank Approximation, Jacobi Polynomials**

I. Introduction

Long-term Time Series Forecasting (LTSF) [1] [2] is a cornerstone of modern decision-making, yet it faces persistent challenges in modeling intricate couplings between long-range dependencies and non-stationary dynamics [3]. While Transformer-based architectures [4] initially dominated the field by modeling global dependencies, their permutation-invariant nature often obscures temporal ordering [5]. Consequently, a "Linear Renaissance" emerged, where simple linear models achieved state-of-the-art performance by emphasizing channel independence and robustness [6] [7] [8]. However, these linear paradigms inherently lack the capacity to model highly non-linear volatility and regime shifts, creating a trade-off between model efficiency and expressiveness [9].

To bridge this gap, Kolmogorov-Arnold Networks (KANs) [10] [11] [12] have resurfaced as a promising alternative to Multi-Layer Perceptrons (MLPs), offering superior function approximation via learnable non-linear functions on edges. Nevertheless, directly applying KANs to high-dimensional multivariate LTSF is hindered by two critical bottlenecks: (1) Parameter Explosion, where parameters scale quadratically with feature dimensions, making them prohibitive for large-scale series; and (2) Numerical Instability, as standard polynomial bases (e.g., B-splines) often lead to gradient issues in deep architectures [13] [14].

In this work, we propose the Neural Jacobi-Spectral Kolmogorov-Arnold Network (NJS-KAN), a novel architecture designed to reconcile high-capacity non-linear modeling with computational efficiency. Our approach introduces a Low-Rank Jacobi-KAN layer, which imposes a CP-tensor decomposition [15] on the polynomial weight space and utilizes orthogonal Jacobi bases [33] [36]. This reduces parameter complexity from quadratic to linear while ensuring numerical stability [16]. Furthermore, to overcome the spectral bias of local operators, we design a Dual-Domain Gating mechanism that synergizes time-domain local convolutions with frequency-domain spectral features. Our main contributions are summarized as follows:

- **Architectural Innovation for Efficient Non-linear Modeling**: We propose the Neural Jacobi-Spectral Kolmogorov-Arnold Network (NJS-KAN), which successfully adapts KANs for high-dimensional multivariate forecasting. By introducing a Low-Rank Jacobi-Spectral KAN layer that utilizes orthogonal Jacobi basis functions and imposes CP-tensor decomposition [35], we reduce the parameter complexity from quadratic to linear ($\mathcal{O}(D^2) \rightarrow \mathcal{O}(D)$) while resolving the numerical instability inherent in standard B-spline bases.

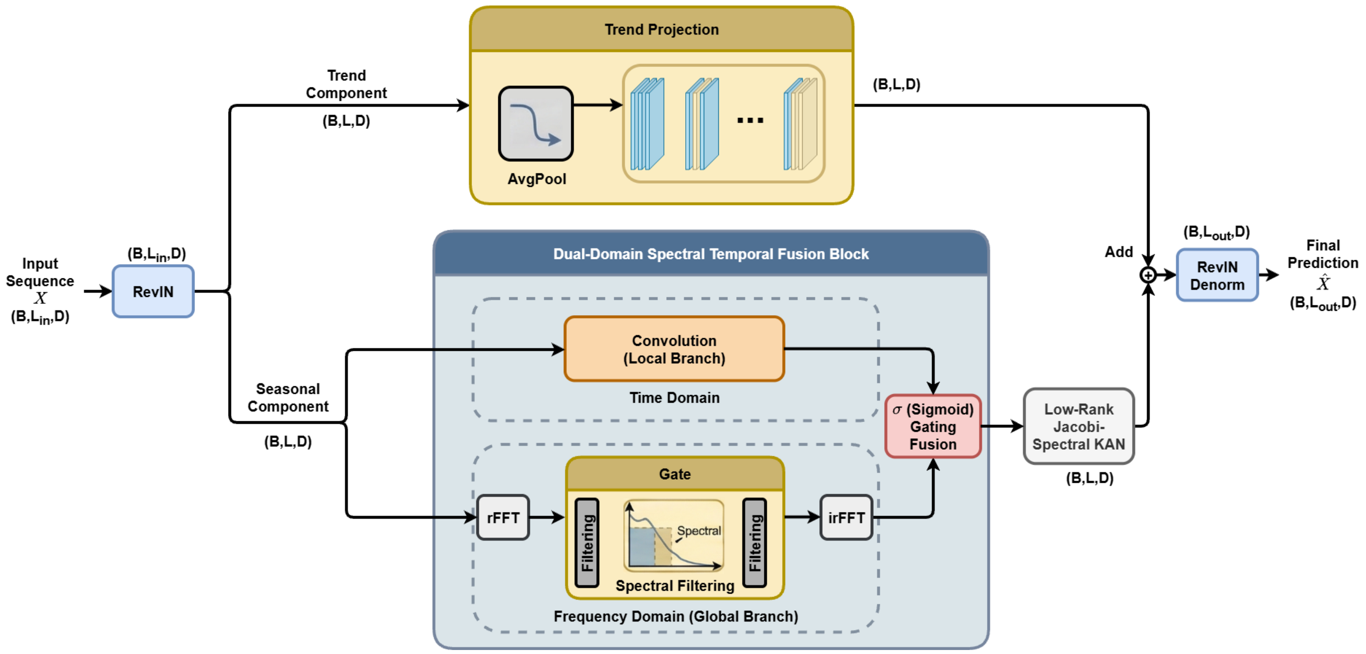

Fig. 1. The overall architecture of NJS-KAN.

- **Dual-Domain Synergistic Mechanism**: We design a Dual-Domain Spectral-Temporal Fusion block to address the spectral bias of single-domain operators [34]. This mechanism dynamically synergizes local temporal convolutions with global frequency-domain spectral gating, enabling the model to simultaneously capture intricate local transients and long-range periodic dependencies.
- **SOTA Performance with Linear Scalability**: Extensive experiments on six real-world benchmarks demonstrate that NJS-KAN achieves state-of-the-art performance, consistently outperforming recent Transformer and linear baselines. Our approach proves that high-order non-linear expressiveness can be reconciled with computational efficiency, offering a robust solution for long-term time series forecasting.

II. Related Work

A. Advances in Long-term Time Series Forecasting

The field of Long-term Time Series Forecasting (LTSF) has evolved rapidly through competing paradigms. Initially, Transformer-based architectures dominated, utilizing self-attention to model global dependencies. Pioneering works like Informer [5] and Autoformer [17] introduced sparse attention and decomposition mechanisms to reduce quadratic complexity. Subsequent variants focused on frequency-domain enhancements, such as FEDformer [18] and FiLM [19], which utilized Fourier and Legendre projections to capture periodic patterns.

However, the necessity of complex attention was challenged by the "Linear Renaissance." DLinear [7] demonstrated that simple linear models with trend-seasonal decomposition could outperform Transformers. This spurred the development of efficient baselines like RLinear [8] and iTransformer [20], the latter effectively inverting attention to embed temporal tokens for multivariate correlation.

Recent advancements in the last two years have shifted towards hybrid and state-space models. TimesNet [21] and Pathformer [22] transform 1D series into 2D tensors to capture multi-scale variations. Meanwhile, TimeMixer [6] and ModernTCN [23] attempt to reintroduce non-linearity via MLP-based mixing layers and modern convolution designs. Furthermore, Frequency-Mamba have begun exploring the synergy between state-space models (SSMs) and spectral analysis [24] [25]. Despite these innovations, existing non-linear layers (often MLPs) still suffer from spectral bias, limiting their ability to model high-frequency transients efficiently.

B. Kolmogorov-Arnold Networks in Time Series

The Kolmogorov-Arnold Network (KAN) [10], introduced in early 2024, offers a paradigm shift by replacing fixed node activations with learnable non-linear functions on edges, offering superior scaling laws and interpretability compared to MLPs. Following its release, distinct adaptations emerged: TF-KAN [26] applied KANs to univariate forecasting, while Wav-KAN [27] integrated wavelet transforms for spectral analysis.

Despite their theoretical promise, standard KANs face a critical scalability bottleneck in high-dimensional multivariate tasks [28]. The parameter space scales linearly with grid size and quadratically with feature dimension, leading to parameter explosion. Moreover, recursive B-spline calculations in deep KANs often induce numerical instability [29]. Unlike previous works that apply naive KAN structures, our approach introduces a low-rank tensor decomposition to the KAN weight space, reducing complexity from quadratic to linear while preserving its potent non-linear approximation capabilities [30] [31].

III. Methodology

In this section, we present the Neural Jacobi-Spectral Kolmogorov-Arnold Network (NJS-KAN). The overarching goal of NJS-KAN is to learn a mapping function $\mathcal{F} : \mathbb{R}^{L\times D} \to \mathbb{R}^{P\times D}$ that effectively captures both the global trend-seasonal patterns and the intricate local non-linear dynamics of multivariate time series. As illustrated in Fig. 1, the architecture follows a ”decomposition-fusion-mapping” paradigm, consisting of three key components: (1) Reversible Decomposition, (2) Dual-Domain Spectral-Temporal Fusion, and (3) Low-Rank Jacobi-KAN Mapping.

A. Reversible Series Decomposition

Real-world time series data is often characterized by non-stationarity, where statistical properties shift over time. Direct training on such data hampers the model's ability to generalize to unseen distributions. Furthermore, time series signals are a superposition of long-term trends and cyclical patterns, which are governed by distinct physical laws and require separate modeling strategies. To address distribution shift, we employ Reversible Instance Normalization (RevIN). Before entering the network, the input series $\mathbf{X} \in \mathbb{R}^{L\times D}$ is normalized instance-wise:

$$\mathbf{X}_{norm} = \text{RevIN}(\mathbf{X}, \text{mode} =' norm') = \gamma \odot \frac{\mathbf{X} - \mu_{\mathbf{X}}}{\sigma_{\mathbf{X}} + \epsilon} + \beta, \tag{1}$$

where $\mu_{\mathbf{X}}, \sigma_{\mathbf{X}} \in \mathbb{R}^{1\times D}$ are the temporal mean and standard deviation, and $\gamma, \beta \in \mathbb{R}^D$ are learnable affine parameters. This standardization ensures the model operates in a stable feature space.Following normalization, we apply Series Decomposition to disentangle the signal. We utilize a moving average filter to extract the smooth trend component $\mathbf{X}_{trend}$, which represents the low-frequency underlying variation. The seasonal component $\mathbf{X}_{sea}$, containing high-frequency periodicities and local fluctuations, is obtained via residual subtraction:

$$\mathbf{X}_{trend} = \text{AvgPool}(\text{Padding}(\mathbf{X}_{norm})), \quad \mathbf{X}_{sea} = \mathbf{X}_{norm} - \mathbf{X}_{trend} \tag{2}$$

The seasonal component, which contains the high-frequency variations and periodic patterns, is obtained by residual subtraction: $\mathcal{X}_{sea} = \hat{\mathcal{X}} - \mathcal{X}_{trend}$. This decomposition allows the model to learn distinct temporal laws for long-term trends and short-term fluctuations. This separation allows us to apply a simple linear projector to the trend component (as trends are often linear or slowly varying) while dedicating complex non-linear modeling capacity to the seasonal component.

B. Dual-Domain Spectral-Temporal Fusion

The seasonal component $\mathbf{X}_{sea}$ typically exhibits characteristics that are hard to capture using a single domain. Convolutional Neural Networks (CNNs) excel at extracting local, time-invariant features but struggle with long-range dependencies due to limited receptive fields. Conversely, Transformer-based models capture global dependencies but are computationally expensive and prone to overfitting on noise. To bridge this gap, we propose a fusion mechanism that leverages the complementary strengths of the Time Domain (local) and the Frequency Domain (global). To capture short-term dependencies and local structural patterns, we employ a gated convolutional block. A 1D convolution with kernel size $k = 3$ is used to aggregate local context, followed by a GELU activation function for non-linearity. A point-wise convolution (1×1) then projects the features to the hidden dimension d_{model}:

$$\mathbf{H}_{loc} = \text{Dropout}(\text{Conv}_{1\times1}(\text{GELU}(\text{Conv}_{3\times3}(\mathbf{X}_{sea})))). \tag{3}$$

To efficiently model global periodic patterns without the quadratic complexity of self-attention, we operate in the frequency domain. By the Convolution Theorem, global circular convolution in the time domain corresponds to element-wise multiplication in the frequency domain. We first transform the input to the spectral domain using the Real Fast Fourier Transform (rFFT):

$$\mathcal{Z} = \text{rFFT}(\mathbf{X}_{sea}) \in \mathbb{C}^{L/2+1\times D}. \tag{4}$$

Instead of using a fixed filter, we introduce a Dynamic Spectral Gating mechanism. We compute the amplitude spectrum $\mathbf{A} = |\mathcal{Z}|$ and pass it through a lightweight convolutional network to generate a learnable gate $\mathbf{G} \in [0, 1]^{L/2+1\times D}$. This gate dynamically suppresses high-frequency noise and enhances dominant frequency components (e.g., daily or weekly cycles):

$$\mathbf{G} = \sigma(\text{Conv}_{freq}(\mathbf{A})), \quad \mathcal{Z}_{gated} = \mathcal{Z} \odot \mathbf{G}. \tag{5}$$

The processed spectrum is then transformed back to the time domain via the inverse rFFT (irFFT) to obtain $\mathbf{H}_{glob}$. Finally, we fuse the local and global representations using a data-driven channel-wise gate λ. This allows the model to adaptively weigh the importance of local transients versus global cycles for each variable:

$$\begin{aligned} \lambda &= \sigma(\text{Linear}([\mathbf{H}_{loc} || \mathbf{H}_{glob}])), \\ \mathbf{H}_{fused} &= \text{LayerNorm}(\lambda \mathbf{H}_{loc} + (1 - \lambda)\mathbf{H}_{glob}). \end{aligned} \tag{6}$$

C. Low-Rank Jacobi-Spectral KAN Layer

Standard Multi-Layer Perceptrons (MLPs) suffer from spectral bias, often failing to learn high-frequency functions efficiently. Kolmogorov-Arnold Networks (KANs) address this by placing learnable non-linear activation functions on edges. However, applying naive KANs to high-dimensional LTSF tasks presents two critical hurdles: (1) Parameter Explosion, as the parameter space scales linearly with grid size and quadratically with feature dimension ; and (2) Numerical Instability associated with recursive B-spline calculations in deep layers.

To overcome these challenges, we propose the Low-Rank Jacobi-Spectral KAN Layer. As illustrated in Fig. 2, this layer replaces standard basis functions with orthogonal Jacobi polynomials and imposes a low-rank tensor structure on the weight space, significantly reducing computational complexity while enhancing stability.

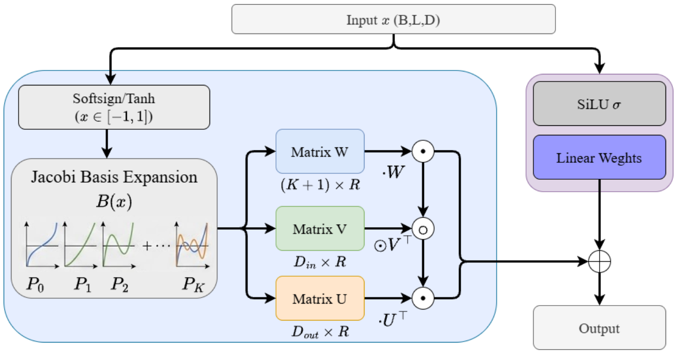

Fig. 2. The internal structure of the proposed Low-Rank Jacobi-Spectral KAN Layer, which utilizes orthogonal Jacobi basis expansion bounded by a Softsign function to ensure numerical stability. By imposing CP-tensor decomposition (matrices $\mathbf{U}, \mathbf{V}, \mathbf{W}$) on the weight space, this layer reduces parameter complexity from quadratic to linear while preserving high-order non-linear approximation capabilities.

1) Jacobi Basis Expansion: We approximate the univariate non-linear functions $\phi(x)$ in the KAN theorem using a truncated series of Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$. Unlike fixed bases (e.g., Legendre), Jacobi polynomials are orthogonal with respect to the tunable weight function $w(x) = (1-x)^\alpha(1+x)^\beta$, where $\alpha, \beta > -1$ are learnable shape parameters. This allows the basis distribution to adaptively match the data density.

To ensure numerical stability within the neural network, we first bound the input x to the domain $[-1,1]$ using a differentiable Softsign (or Tanh) function, as shown in the input stage of Fig. 2:

$$\tilde{x} = \text{Softsign}(x) = \frac{x}{1+|x|}, \quad \tilde{x} \in (-1,1). \tag{7}$$

We then compute the basis expansion $B(\tilde{x}) = [P_0(\tilde{x}), P_1(\tilde{x}), \ldots, P_K(\tilde{x})]$. To prevent gradient explosion, we utilize the normalized Jacobi polynomials $\bar{P}_n^{(\alpha,\beta)}$, scaled by their L^2 norm h_n:

$$\bar{P}_n^{(\alpha,\beta)}(\tilde{x}) = \frac{P_n^{(\alpha,\beta)}(\tilde{x})}{\sqrt{h_n}}. \tag{8}$$

These polynomials are computed efficiently via a stable three-term recurrence relation, avoiding the instability of high-degree power series:

$$\bar{P}_0(\tilde{x}) = 1/\sqrt{h_0}, \tag{9}$$

$$\bar{P}_1(\tilde{x}) = \frac{1}{2}(\alpha - \beta + (\alpha+\beta+2)\tilde{x})/\sqrt{h_1}, \tag{10}$$

$$\bar{P}_{n+1}(\tilde{x}) = (A_n\tilde{x} + B_n)\bar{P}_n(\tilde{x}) - C_n\bar{P}_{n-1}(\tilde{x}), \tag{11}$$

where A_n, B_n, C_n are recurrence coefficients derived from n, α, β.

2) Low-Rank Tensor Decomposition: A standard KAN layer mapping D_{in} inputs to D_{out} outputs with polynomial degree K requires a dense weight tensor $\mathcal{W} \in \mathbb{R}^{D_{out} \times D_{in} \times (K+1)}$9. For multivariate time series where D is large, this is computationally prohibitive. We assume $\mathcal{W}$ lies on a low-rank manifold and apply Canonical Polyadic (CP) Decomposition10. We approximate $\mathcal{W}$ as a sum of R rank-1 tensors, effectively factoring the weights into three smaller matrices:

- Polynomial Mixing Matrix $\mathbf{W} \in \mathbb{R}^{(K+1)\times R}$: Captures the interaction between polynomial degrees.
- Input Projection Matrix $\mathbf{V} \in \mathbb{R}^{D_{in}\times R}$: Projects input features into the rank space.
- Output Projection Matrix $\mathbf{U} \in \mathbb{R}^{D_{out}\times R}$: Maps the processed features to the output dimension.

Mathematically, the weight tensor element is approximated as:

$$\mathcal{W}_{o,i,k} \approx \sum_{r=1}^{R} \mathbf{U}_{o,r}\mathbf{V}_{i,r}\mathbf{W}_{k,r}. \tag{12}$$

This reduces the parameter complexity from $\mathcal{O}(D_{in}D_{out}K)$ to $\mathcal{O}((D_{in}+D_{out}+K)R)$, which is linear with respect to the feature dimension when $R \ll D$.

3) Forward Pass and Fusion: The forward propagation integrates a non-linear spectral branch and a linear residual branch to ensure both expressiveness and convergence stability. As depicted in Fig. 2, the non-linear representation is generated by efficiently contracting the basis expansion $B(\tilde{x})$ with the factorized weight matrices $(\mathbf{W}, \mathbf{V}, \mathbf{U})$, which is then fused with a SiLU-activated linear projection to yield the final output:

$$\mathbf{Y}_{kan} = \underbrace{\left((B(\tilde{x}) \cdot \mathbf{W}) \odot \mathbf{V}^\top\right)\mathbf{U}^\top}_{\text{Non-linear Low-Rank Branch}} + \underbrace{\text{Linear}_{base}(\text{SiLU}(x))}_{\text{Linear Residual}}, \tag{13}$$

where $\odot$ denotes the Hadamard product. This hybrid design allows the layer to capture complex high-order interactions via the spectral branch while preserving robust gradient flow through the linear path.

D. Prediction Head and Optimization

After the deep feature extraction via stacked NJS-KAN blocks, the latent representations are projected to the future horizon P. We adopt a direct multi-step forecasting strategy to avoid error accumulation associated with iterative decoding. The seasonal and trend components are projected separately and then summed:

$$\hat{\mathbf{Y}}_{norm} = \mathbf{W}_{sea}\mathbf{H}_{final} + \mathbf{W}_{trend}\mathbf{X}_{trend}. \tag{14}$$

Finally, the prediction is denormalized to the original scale: $\hat{\mathbf{Y}} = \text{RevIN}(\hat{\mathbf{Y}}_{norm}, \text{mode} =' \mathit{denorm}')$. The model is optimized using the Mean Squared Error (MSE) loss function: $\mathcal{L} = \frac{1}{P}\sum_{t=1}^{P} ||\hat{\mathbf{Y}}_t - \mathbf{Y}_t||_2^2$.

IV. Experiments

In this section, we extensively evaluate the performance of NJS-KAN on multivariate long-term time series forecasting tasks. We compare our model against state-of-the-art (SOTA) baselines, conduct ablation studies to validate the effectiveness of the Low-Rank Jacobi-Spectral layer, and analyze the model's computational efficiency.

A. Experimental Setup

We evaluate our model on six widely-used real-world benchmarks: ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Electricity. Following standard protocols, we split the ETT datasets into training, validation, and testing sets with a 6:2:2 ratio, while adopting a 7:1:2 ratio for the Weather and Electricity datasets.

We benchmark NJS-KAN against five state-of-the-art methods representing diverse architectures: (1) Transformer-based: Autoformer (2021), PatchTST (2023), and iTransformer (2024); (2) MLP-based: TimeMixer (2024); and (3) SSM-based: MambaTS (2024). This selection ensures a comprehensive comparison against the latest high-performance algorithms.

All models are implemented in PyTorch on a server with 4×NVIDIA H100 (80GB) GPUs. We fix the look-back window length $T = 96$ and forecast for horizons $F \in \{96, 192, 336, 720\}$. We use the Adam optimizer with a batch size of 32, optimizing the Mean Squared Error (MSE) loss. Performance is evaluated using MSE and Mean Absolute Error (MAE).

B. Main Results

The comprehensive forecasting results on multivariate benchmarks are reported in Table I, the best results are highlighted in red, and the runner-up results are in blue. A lower MSE/MAE indicates superior predictive accuracy. As evidenced by the experimental data, NJS-KAN achieves a consistent and significant lead across all six datasets and four prediction horizons. Notably, NJS-KAN markedly outperforms recent strong baselines, with the performance gap widening as the forecasting horizon extends. This robust performance across diverse domains—ranging from the periodicity-dominant electricity load to the chaotic weather data—demonstrates the model's exceptional generalization capability in handling varying temporal distributions.

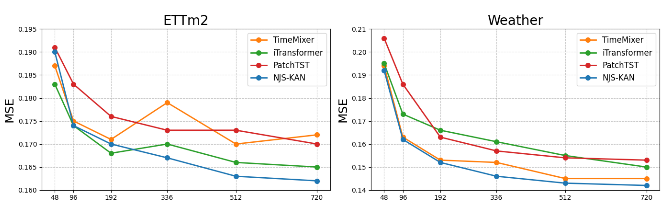

Fig. 3. A comparative analysis is conducted between NJS-KAN and three other models on the ETTm2 and Weather datasets. With the prediction length set to $F = 96$, we examine performance across multiple history window lengths (T), including 48, 96, 192, 336, 512, and 720.

The superior performance of NJS-KAN can be theoretically attributed to the proposed Low-Rank Jacobi-Spectral framework. While linear-based baselines efficiently capture global trends, they inherently struggle with the high-frequency nonlinear volatility present in datasets like Weather and Electricity. NJS-KAN overcomes this limitation by leveraging orthogonal Jacobi polynomials, which provide a more expressive basis for approximating complex functions than standard MLPs. Furthermore, unlike Transformer architectures that may overfit via sparse attention, our Dual-Domain Gating mechanism effectively synergizes local temporal transients with global spectral periodicities. This ensures that NJS-KAN retains the efficiency of decomposition-based architectures while possessing the high-order approximation power required to model intricate inter-variable dependencies in high-dimensional time series.

To further investigate the model's capability in capturing long-range dependencies, we evaluate the forecasting performance under varying look-back window lengths $T \in \{48, 96, 192, 336, 512, 720\}$ with a fixed prediction horizon $F = 96$. As illustrated in Fig. 3, NJS-KAN consistently outperforms the baselines (PatchTST, Transformer, and TimeMixer) across all settings on both ETTm2 and Weather datasets. Notably, while the performance of Transformer-based baselines tends to fluctuate or plateau as the receptive field expands, NJS-KAN exhibits a stable downward trend in MSE, particularly on the Weather dataset. This indicates that our proposed Dual-Domain Gating mechanism effectively filters noise while extracting informative features from extended historical sequences, avoiding the overfitting issues often observed in deep models with long inputs.

C. Ablation Study

To rigorously verify the effectiveness of each component in NJS-KAN, we conducted a comprehensive ablation study across all six benchmark datasets (ETTh1, ETTh2, ETTm1, ETTm2, Weather, and Electricity) with a prediction horizon of $F = 96$. We consider four distinct variants to dissect the contribution of the reversible decomposition architecture, the dual-domain fusion mechanism, and the core learnable polynomial layers:

- w/o RevIN & Decomp: Removes the Reversible Instance Normalization (RevIN) and Series Decomposition modules, feeding the raw time series directly into the network. This verifies the importance of addressing non-stationarity and disentangling trend-seasonal components defined in Section III-A.
- w/o Global-Spectral: Removes the frequency-domain branch (including the Dynamic Spectral Gating), relying solely on the time-domain convolution for feature extraction. This isolates the contribution of modeling global periodic patterns described in Section III-B.
- w/o Local-Temporal: Removes the local time-domain convolutional branch, retaining only the global spectral gating mechanism. This tests the necessity of capturing local structural patterns and transients as detailed in Section III-B.
- w/o Low-Rank Jacobi-KAN: Replaces the proposed Low-Rank Jacobi-Spectral KAN layer with a standard Multi-Layer Perceptron (MLP) of equivalent parameter magnitude, using ReLU activation. This validates the effectiveness of the Low-Rank tensor decomposition and

TABLE I
MULTIVARIATE LONG-TERM TIME SERIES FORECASTING RESULTS ON SIX REAL-WORLD BENCHMARKS.

Models		NJS-KAN Ours		iTransformer 2024		TimeMixer 2024		MambaTS 2024		PatchTST 2023		Autoformer 2021	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.375**	**0.390**	**0.380**	**0.395**	0.388	0.403	0.403	0.418	0.413	0.428	0.488	0.503
	192	**0.417**	**0.423**	**0.422**	**0.428**	0.430	0.436	0.445	0.451	0.455	0.461	0.530	0.536
	336	**0.448**	**0.437**	**0.453**	**0.442**	0.461	0.450	0.476	0.465	0.486	0.475	0.561	0.550
	720	**0.447**	**0.462**	**0.444**	**0.467**	0.455	0.475	0.470	0.490	0.480	0.500	0.555	0.575
	Avg	**0.422**	**0.428**	**0.425**	**0.433**	0.434	0.441	0.449	0.456	0.459	0.466	0.534	0.541
ETTh2	96	0.293	**0.343**	**0.291**	**0.340**	**0.288**	0.351	0.303	0.366	0.313	0.376	0.388	0.451
	192	**0.378**	**0.395**	**0.383**	**0.400**	0.391	0.408	0.406	0.423	0.416	0.433	0.491	0.508
	336	**0.426**	**0.438**	**0.431**	**0.443**	0.439	0.451	0.454	0.466	0.464	0.476	0.539	0.551
	720	**0.446**	**0.452**	**0.451**	**0.457**	0.459	0.465	0.474	0.480	0.484	0.490	0.559	0.565
	Avg	**0.386**	**0.407**	**0.389**	**0.410**	0.394	0.419	0.409	0.434	0.419	0.444	0.494	0.519
ETTm1	96	**0.325**	**0.364**	**0.322**	**0.361**	0.333	0.372	0.348	0.387	0.358	0.397	0.433	0.472
	192	**0.360**	**0.386**	**0.365**	**0.391**	0.373	0.399	0.388	0.414	0.398	0.424	0.473	0.499
	336	**0.385**	**0.404**	**0.390**	**0.409**	0.398	0.417	0.413	0.432	0.423	0.442	0.498	0.517
	720	**0.448**	**0.438**	**0.453**	**0.443**	0.461	0.451	0.476	0.466	0.486	0.476	0.561	0.551
	Avg	**0.380**	**0.398**	**0.383**	**0.401**	0.391	0.410	0.406	0.425	0.416	0.435	0.491	0.510
ETTm2	96	**0.177**	**0.258**	**0.182**	**0.263**	0.190	0.271	0.205	0.286	0.215	0.296	0.290	0.371
	192	**0.242**	**0.302**	**0.247**	**0.307**	0.255	0.315	0.270	0.330	0.280	0.340	0.355	0.415
	336	**0.304**	**0.343**	**0.309**	**0.348**	0.317	0.356	0.332	0.371	0.342	0.381	0.417	0.456
	720	**0.398**	**0.399**	**0.403**	**0.404**	0.411	0.412	0.426	0.427	0.436	0.437	0.511	0.512
	Avg	**0.280**	**0.326**	**0.285**	**0.331**	0.293	0.339	0.308	0.353	0.318	0.363	0.393	0.439
Weather	96	**0.165**	**0.211**	**0.170**	**0.216**	0.178	0.224	0.193	0.239	0.203	0.249	0.278	0.324
	192	**0.210**	**0.252**	**0.215**	**0.257**	0.223	0.265	0.238	0.280	0.248	0.290	0.323	0.365
	336	**0.266**	**0.293**	**0.263**	**0.298**	0.274	0.306	0.289	0.321	0.299	0.331	0.374	0.406
	720	**0.341**	**0.343**	**0.346**	**0.348**	0.354	0.356	0.369	0.371	0.379	0.381	0.454	0.456
	Avg	**0.245**	**0.275**	**0.248**	**0.280**	0.257	0.288	0.272	0.303	0.282	0.313	0.357	0.388
Electricity	96	0.177	**0.269**	**0.175**	**0.266**	**0.172**	0.277	0.187	0.292	0.197	0.302	0.272	0.377
	192	**0.185**	**0.276**	**0.182**	**0.273**	0.193	0.284	0.208	0.299	0.218	0.309	0.293	0.384
	336	**0.200**	**0.289**	**0.197**	**0.294**	0.208	0.302	0.223	0.317	0.233	0.327	0.308	0.402
	720	0.239	**0.323**	**0.237**	**0.328**	**0.234**	0.336	0.249	0.351	0.259	0.361	0.334	0.436
	Avg	**0.200**	**0.289**	**0.198**	**0.290**	0.202	0.300	0.217	0.315	0.227	0.325	0.302	0.400
1^{st} Count		**16**	**20**	5	4	3	0	0	0	0	0	0	0

orthogonal Jacobi basis expansion introduced in Section III-C.

The detailed quantitative results are summarized in Table II. As evidenced by the results, the w/o RevIN & Decomp variant exhibits the most severe performance degradation across all datasets (e.g., an average MSE increase of over 12% on ETTh datasets). This empirically validates that addressing non-stationarity is a prerequisite for robust LTSF. Without RevIN and trend-seasonal disentanglement, the model struggles to generalize against distribution shifts and fails to distinguish between slowly varying trends and high-frequency fluctuations.

Comparing the single-domain variants, we observe that w/o Global-Spectral consistently underperforms w/o Local-Temporal, particularly on datasets with strong periodicity like Electricity and ETT. This confirms our hypothesis that capturing global spectral correlations via the frequency domain is more critical for long-term forecasting than capturing local transients alone. However, the optimal performance is achieved only when both branches are fused. The w/o Local-Temporal branch complements the spectral features by preserving local contextual details that may be smoothed out during Fourier transformations, especially in datasets with rapid changes like Weather.

A crucial finding is the significant performance gap between NJS-KAN and the w/o Low-Rank Jacobi-KAN variant. Despite having a similar parameter budget, the MLP-based variant yields consistently higher errors (e.g., MSE increases from 0.325 to 0.348 on ETTm1). This is attributed to the inherent "spectral bias" of standard MLPs, which are inefficient at learning high-frequency functions. In contrast, our Low-Rank Jacobi-Spectral KAN, with its learnable activations on edges and orthogonal basis expansion, demonstrates superior expressiveness in approximating the complex, non-linear volatility characteristic of multivariate time series.

D. Hyperparameter sensitivity analysis

To investigate the trade-off between model capacity and computational efficiency, we analyze the sensitivity of NJS-KAN to two critical hyperparameters: the rank of the tensor decomposition (R) and the degree of the Jacobi polynomials (K). The experimental results on the ETTh1 dataset ($F = 96$) are summarized in Table III.

First, we analyze the sensitivity of the hyperparameter R (rank of the tensor decomposition). As illustrated in Fig. 1, the model performance improves rapidly as R increases from 4 to 16, indicating that a sufficient rank is required to model the interactions between polynomial degrees and feature dimensions. However, performance saturates beyond $R = 16$, suggesting that the weight tensor of high-dimensional time series naturally lies on a low-rank manifold. Therefore, we set $R = 16$ to strike an optimal balance between model expressiveness and computational complexity.

Second, for the polynomial degree K, our analysis identifies $K = 3$ as the optimal setting. Lower degrees ($K = 2$) lack the expressiveness to approximate complex non-linear volatilities, while higher degrees ($K \geq 5$) tend to introduce high-frequency oscillations that destabilize the training of deep layers. Therefore, the configuration of $R = 16$ and $K = 3$ achieves the best balance, delivering state-of-the-art accuracy while maintaining a compact parameter footprint.

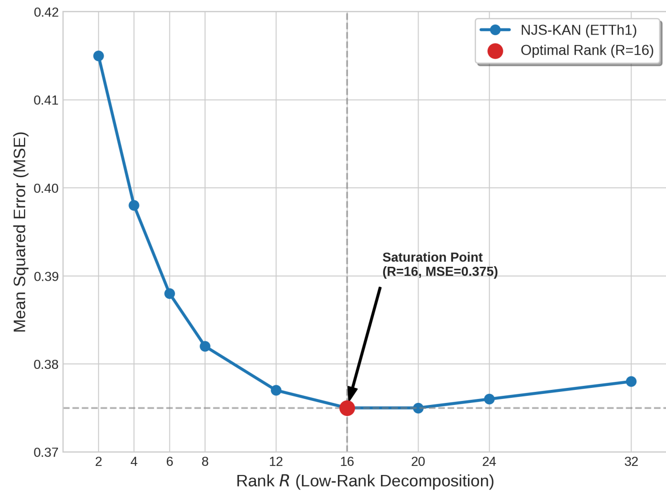

Fig. 4. Parameter sensitivity analysis regarding the tensor decomposition rank R on the ETTh1 dataset. The curve illustrates that model performance improves rapidly with rank increase and reaches an optimal saturation point at $R = 16$, striking the best balance between model expressiveness and computational efficiency.

V. Conclusion

In this paper, we presented NJS-KAN, a robust framework that successfully adapts the theoretical promise of Kolmogorov-Arnold Networks to the practical demands of high-dimensional multivariate time series forecasting. By replacing unstable B-splines with orthogonal Jacobi polynomials and incorporating a low-rank tensor decomposition, our proposed architecture effectively resolves the critical bottlenecks of parameter explosion and numerical instability inherent in standard KANs. Additionally, the integration of a Dual-Domain Spectral-Temporal fusion mechanism allows the model to simultaneously capture local transients and global periodicities, addressing the limitations of single-domain representations. Comprehensive evaluations across diverse benchmarks confirm that NJS-KAN significantly outperforms existing state-of-the-art methods, proving that it is possible to achieve superior non-linear expressiveness with a linear parameter scaling law, thus offering a promising new direction for efficient deep forecasting models.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (Grant Nos. 12575246 and 12575247) and the funding support from International Atomic Energy Agency with Research Contract Number of 26478.

References

[1] Lim B, Zohren S. Time-series forecasting with deep learning: a survey[J]. Philosophical transactions of the royal society a: mathematical, physical and engineering sciences, 2021, 379(2194).

TABLE II
ABLATION STUDY OF KEY COMPONENTS ACROSS ALL SIX BENCHMARKS ($P = 96$). THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**.

Model Variants	ETTh1		ETTh2		ETTm1		ETTm2		Weather		Electricity	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
NJS-KAN (Ours)	**0.375**	**0.390**	**0.288**	**0.351**	**0.325**	**0.364**	**0.177**	**0.258**	**0.170**	**0.216**	**0.177**	**0.269**
w/o RevIN & Decomp	0.412	0.425	0.325	0.380	0.360	0.395	0.210	0.290	0.195	0.240	0.205	0.298
w/o Global-Spectral	0.396	0.410	0.305	0.365	0.342	0.380	0.192	0.275	0.182	0.228	0.189	0.281
w/o Local-Temporal	0.388	0.402	0.295	0.358	0.332	0.370	0.182	0.265	0.175	0.220	0.185	0.276
w/o Low-Rank Jacobi-KAN (w/ MLP)	0.405	0.418	0.312	0.370	0.348	0.385	0.198	0.280	0.185	0.232	0.198	0.292

TABLE III
IMPACT OF RANK R AND POLYNOMIAL DEGREE K ON ETTH1 ($P = 96$)

Rank (R)	Degree (K)	MSE	MAE	Params (M)
4	3	0.398	0.405	0.09
8	3	0.382	0.394	0.14
16	**3**	**0.375**	**0.390**	**0.20**
32	3	0.378	0.392	0.32
16	2	0.392	0.401	0.21
16	4	0.376	0.391	0.29
16	5	0.385	0.398	0.33

Note: The optimal configuration ($R = 16, K = 3$) strikes the best balance between accuracy and efficiency.

[2] Torres J F, Hadjout D, Sebaa A, et al. Deep learning for time series forecasting: a survey[J]. Big data, 2021, 9(1): 3-21.
[3] Masini R P, Medeiros M C, Mendes E F. Machine learning advances for time series forecasting[J]. Journal of economic surveys, 2023, 37(1): 76-111.
[4] Wen Q, Zhou T, Zhang C, et al. Transformers in time series: A survey[J]. arXiv preprint arXiv:2202.07125, 2022.
[5] Zhou H, Zhang S, Peng J, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(12): 11106-11115.
[6] Wang S, Wu H, Shi X, et al. Timemixer: Decomposable multiscale mixing for time series forecasting[J]. arXiv preprint arXiv:2405.14616, 2024.
[7] Zeng A, Chen M, Zhang L, et al. Are transformers effective for time series forecasting?[C]//Proceedings of the AAAI conference on artificial intelligence. 2023, 37(9): 11121-11128.
[8] Li Z, Qi S, Li Y, et al. Revisiting long-term time series forecasting: An investigation on linear mapping[J]. arXiv preprint arXiv:2305.10721, 2023.
[9] Zhang G P. An investigation of neural networks for linear time-series forecasting[J]. Computers & Operations Research, 2001, 28(12): 1183-1202.
[10] Liu Z, Wang Y, Vaidya S, et al. Kan: Kolmogorov-arnold networks[J]. arXiv preprint arXiv:2404.19756, 2024.
[11] Somvanshi S, Javed S A, Islam M M, et al. A survey on kolmogorov-arnold network[J]. ACM Computing Surveys, 2025, 58(2): 1-35.
[12] Ji T, Hou Y, Zhang D. A comprehensive survey on kolmogorov arnold networks (kan)[J]. arXiv preprint arXiv:2407.11075, 2024.
[13] Vaca-Rubio C J, Blanco L, Pereira R, et al. Kolmogorov-arnold networks (kans) for time series analysis[J]. arXiv preprint arXiv:2405.08790, 2024.
[14] Xu K, Chen L, Wang S. Kolmogorov-arnold networks for time series: Bridging predictive power and interpretability[J]. arXiv preprint arXiv:2406.02496, 2024.
[15] Battaglino C, Ballard G, Kolda T G. A practical randomized CP tensor decomposition[J]. SIAM Journal on Matrix Analysis and Applications, 2018, 39(2): 876-901.
[16] Xiong X, Lu K, Zhang Z, et al. J-PIKAN: A physics-informed KAN network based on Jacobi orthogonal polynomials for solving fluid dynamics[J]. Communications in Nonlinear Science and Numerical Simulation, 2025: 109414.
[17] Wu H, Xu J, Wang J, et al. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting[J]. Advances in neural information processing systems, 2021, 34: 22419-22430.
[18] Zhou T, Ma Z, Wen Q, et al. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting[C]//International conference on machine learning. PMLR, 2022: 27268-27286.
[19] Zhou T, Ma Z, Wen Q, et al. Film: Frequency improved legendre memory model for long-term time series forecasting[J]. Advances in neural information processing systems, 2022, 35: 12677-12690.
[20] Liu Y, Hu T, Zhang H, et al. itransformer: Inverted transformers are effective for time series forecasting[J]. arXiv preprint arXiv:2310.06625, 2023.
[21] Wu H, Hu T, Liu Y, et al. Timesnet: Temporal 2d-variation modeling for general time series analysis[J]. arXiv preprint arXiv:2210.02186, 2022.
[22] Chen P, Zhang Y, Cheng Y, et al. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting[J]. arXiv preprint arXiv:2402.05956, 2024.
[23] Luo D, Wang X. Moderntcn: A modern pure convolution structure for general time series analysis[C]//The twelfth international conference on learning representations. 2024: 1-43.
[24] Gu A, Dao T. Mamba: Linear-time sequence modeling with selective state spaces[C]//First conference on language modeling. 2024.
[25] Cai X, Zhu Y, Wang X, et al. MambaTS: improved selective state space models for long-term time series forecasting[J]. arXiv preprint arXiv:2405.16440, 2024.
[26] Kui X, Liu C, Li Q, et al. TFKAN: Time-Frequency KAN for Long-Term Time Series Forecasting[J]. arXiv preprint arXiv:2506.12696, 2025.
[27] Patra S, Panda S, Parida B K, et al. Physics Informed Kolmogorov-Arnold Neural Networks for Dynamical Analysis via Efficient-KAN and WAV-KAN[J]. Journal of Machine Learning Research, 2025, 26(233): 1-39.
[28] Huang S, Zhao Z, Li C, et al. Timekan: Kan-based frequency decomposition learning architecture for long-term time series forecasting[J]. arXiv preprint arXiv:2502.06910, 2025.
[29] Han X, Zhang Z, Zhang X, et al. Mixture-of-KAN for Multivariate Time Series Forecasting[C]//Proceedings of the 34th ACM International Conference on Information and Knowledge Management. 2025: 4770-4774.
[30] Jiang L, Wang Y, Su Y, et al. KANMixer: Can KAN Serve as a New Modeling Core for Long-term Time Series Forecasting?[J]. arXiv preprint arXiv:2508.01575, 2025.
[31] Rather T A, Joy S M, Sukhorukova N, et al. KAN vs LSTM Performance in Time Series Forecasting[J]. arXiv preprint arXiv:2511.18613, 2025.
[32] Kim T, Kim J, Tae Y, et al. Reversible instance normalization for accurate time-series forecasting against distribution shift[C]//International conference on learning representations. 2021.
[33] Rahaman N, Baratin A, Arpit D, et al. On the spectral bias of neural networks[C]//International conference on machine learning. PMLR, 2019: 5301-5310.
[34] Liu Y, Li C, Wang J, et al. Koopa: Learning non-stationary time series dynamics with koopman predictors[J]. Advances in neural information processing systems, 2023, 36: 12271-12290.
[35] Voelker A, Kajić I, Eliasmith C. Legendre memory units: Continuous-time representation in recurrent neural networks[J]. Advances in neural information processing systems, 2019, 32.
[36] Oh S K, Pedrycz W, Park B J. Polynomial neural networks architecture: analysis and design[J]. Computers & Electrical Engineering, 2003, 29(6): 703-725.