2025 IEEE International Conference on Future Machine Learning and Data Science (FMLDS) | 979-8-3315-5397-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/FMLDS67896.2025.00034

Spectral Graph Filtering for Long-Term Multivariate Time Series Forecasting

1st Zongyao Feng
Department of Computer and Information Systems
University of Aizu
Aizu, Japan
d8252101@u-aizu.ac.jp

2nd Konstantin Markov
Department of Computer and Information Systems
University of Aizu
Aizu, Japan
markov@u-aizu.ac.jp

***Abstract*—Long-term multivariate time series forecasting requires modeling complex spatiotemporal dependencies, yet existing approaches tend to focus on addressing the problem from the temporal perspective using channel-independent modeling, with few works exploring effective spatial modeling for long-term horizons. While recent methods use attention mechanisms to capture spatial interactions, they suffer from producing diffuse patterns when the number of variables increases. We propose a spectral graph filtering approach that transforms learned attention weights into graph Laplacians, enabling frequency-aware spatial modeling through Chebyshev polynomial approximations. This design treats the multivariate time series as signals over a graph, allowing the model to adaptively learn spatial patterns most critical for long-term forecasting through learnable spectral filters. Our method first projects temporal sequences into a latent space, computes attention-based adjacency matrices, then applies multi-scale spectral filtering to extract predictive spatial patterns. Experiments on three benchmarks, Electricity, Solar and Weather, demonstrate consistent improvements over the representative baselines, validate that frequency-domain spatial modeling offers a principled and scalable alternative to conventional attention mechanisms for long-term multivariate forecasting.**

***Index Terms*—multivariate time series forecasting, spectral graph filtering, spatiotemporal modeling, long-term forecasting, graph signal processing**

I. Introduction

Long-term multivariate time series forecasting (MTSF) is essential for strategic planning and resource allocation across domains including energy management, climate modeling, and financial analytics [1]–[3]. The fundamental challenge lies in accurately predicting future values for multiple interrelated variables over extended horizons, a task complicated by entangled spatiotemporal dependencies that become increasingly noisy and unreliable as the forecasting window extends [4].

Recent advances in long-term MTSF have primarily addressed challenges from the temporal modeling perspective. In particular, linear models have shown strong performance by effectively capturing dominant trends and periodicities in the input sequence, while maintaining computational efficiency [5], [6]. However, these models often overlook the critical role of spatial (i.e., inter-variable) dependencies in MTSF. Typically, channel-independent (CI) modeling [7] is adapted as a workaround, which processes each variate separately. Although this enhances modeling flexibility, it introduces substantial computational overhead and fails to exploit potential cross-channel correlations [6]. Alternatively, attention-based architectures such as iTransformer [8] learn global inter-variable relationships through attention mechanism. While this approach offers more flexibilities, it scales poorly when the number of variates grows, often resulting in diffuse or uninformative attention distributions [9], [10]. These limitations underscore the need for more effective and scalable spatial modeling mechanisms tailored to long-term MTSF.

A promising direction lies in spectral graph filtering [11], which models spatial dependencies in the frequency domain. Instead of learning direct pairwise interactions, spectral filtering operates on the eigenbasis of the graph Laplacian, allowing decomposition of graph signals into frequency components. This enables the model to distinguish between low-frequency structures representing smooth, global correlations and high-frequency components encoding localized, rapidly changing interactions. By leveraging a bank of learnable filters across different frequency bands, e.g. via Chebyshev polynomial filters, spectral filtering captures multi-scale spatial dependencies that are critical for robust long-term forecasting. These frequency-aware capabilities offer a compelling alternative to conventional spatial modeling techniques, particularly in high-dimensional multivariate settings.

To this end, we propose a novel spectral filtering model for MTSF. Our method begins by projecting the temporal dimension into a high-dimensional latent space, after which attention weights are computed across the spatial dimension. These weights are subsequently transformed into a global graph Laplacian matrix to enable graph filtering in the frequency domain. This design allows the model to adaptively learn spatial patterns that are most predictive for long-term horizons. We evaluated our method on three benchmark datasets including electricity, solar energy and weather condition, demonstrating its superiority over representative models. Our module achieves consistent gains in both MAE and RMSE, verifying the effectiveness of our spectral strategy. The main contributions of this work can be summarized as follows:

- We introduce a spectral filtering module that leverages graph signal processing to model spatial dependencies, offering a scalable alternative to attention mechanisms.

979-8-3315-5397-5/25/$31.00 ©2025 IEEE
DOI 10.1109/FMLDS67896.2025.00034

- Our approach shows consistent improvements over representative baselines on three real-world benchmarks, validating the effectiveness of our approach.

II. Related Works

Recent advances in long-term time series forecasting can be broadly categorized based on how they handle spatial dependencies. A prominent line of work adopts CI modeling, where each univariate time series is processed separately using shared model weights. This strategy simplifies the learning process and avoids interference between channels with different periodicities. Autoformer [4] introduces a decomposition block that progressively separates trend and seasonal components throughout the model, and replaces standard attention with Auto-Correlation to discover periodic dependencies. FEDformer [12] integrates seasonal-trend decomposition with frequency-enhanced Transformer blocks, shifting attention computation from the time domain to the frequency domain via randomly selected Fourier or Wavelet components. This design reduces computation complexity while improving the capture of global temporal patterns. DLinear [5] and RLinear [6] demonstrate that linear mappings, when combined with trend-seasonal decomposition or reversible normalization, can effectively capture long-range periodic behavior. PatchTST [13] segments input sequences into local patches and applies a shared Transformer to each channel independently, improving efficiency and generalization. TiDE [14] uses a lightweight MLP encoder-decoder with residual connections and global skip pathways, enabling fast training and strong performance without attention mechanisms. While efficient, these models lack mechanisms for learning spatial dependencies, which is critical for MTSF tasks.

A smaller subset of models explicitly addresses spatial modeling across variables. TimesNet [15] transforms 1D time series into 2D representations based on discovered periodicities, then applies multi-scale inception blocks to capture both intra-periodic and inter-variable patterns. Crossformer [16] introduces a two-stage attention mechanism that separately models cross-time and cross-variable dependencies using a hierarchical encoder-decoder structure. iTransformer [8] reorders the modeling paradigm by treating each variate as a token and performing attention across channels, enabling explicit learning of spatial correlations while temporal patterns are modeled via linear mapping.

III. Preliminaries

a) Graph Representation and Laplacian: Let a weighted directed graph be denoted as $G = (V, E, W)$, where V is the set of N nodes (e.g., variables in a multivariate time series), $E \subseteq V \times V$ is the set of edges, and $W \in \mathbb{R}^{N\times N}$ is the weighted adjacency matrix. The entry $W_{ij} > 0$ indicates the strength of interaction from node i to node j. To support spectral analysis on directed graphs, we adopt the random walk normalized Laplacian [18] defined as:

$$L = I - D^{-1}W, \tag{1}$$

where $D \in \mathbb{R}^{N\times N}$ is the diagonal out-degree matrix with entries $D_{ii} = \sum_j W_{ij}$, and I is the identity matrix. This formulation preserves the stochastic semantics of node transitions and supports spectral graph analysis in directed settings.

b) Graph Spectral Theory: Spectral graph theory studies the structural properties of a graph through the eigendecomposition of its Laplacian. Given L, we compute:

$$L = U\Lambda U^\top, \tag{2}$$

where $U \in \mathbb{R}^{N\times N}$ contains the orthonormal eigenvectors of L, and $\Lambda \in \mathbb{R}^{N\times N}$ is a diagonal matrix of non-negative eigenvalues. The columns of U form the graph Fourier basis, and the eigenvalues in Λ represent graph frequencies. Low-frequency components correspond to smooth, globally coherent patterns, while high-frequency components capture sharp, localized variations.

c) Graph Spectral Filtering: Let $x \in \mathbb{R}^N$ denote a graph signal (e.g., values across all variables at a single time step). Its graph Fourier transform is:

$$\hat{x} = U^\top x, \tag{3}$$

and the inverse transform is:

$$x = U\hat{x}. \tag{4}$$

Given a spectral filter $g(\Lambda)$, the filtered signal is:

$$y = Ug(\Lambda)U^\top x. \tag{5}$$

This operation modifies the signal in the frequency domain, allowing selective emphasis or suppression of different frequency bands. However, it is computationally expensive for large graphs and is not localized in space [19], making this approach impractical for deep learning applications.

d) Chebyshev Polynomial Approximation: To alleviate the computational burden of full spectral filtering, the filter function $g(\Lambda)$ can be approximated using Chebyshev polynomials [19]. First, the Laplacian is rescaled so that its eigenvalues lie within $[-1, 1]$:

$$\tilde{L} = \frac{2L}{\lambda_{\max}} - I, \tag{6}$$

where $\lambda_{\max}$ is the largest eigenvalue of L. The Chebyshev polynomials $T_k(x)$ are defined recursively as:

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x). \tag{7}$$

Using this basis, the filter is approximated as follows:

$$g(\Lambda) \approx \sum_{k=0}^{K} \theta_k T_k(\tilde{\Lambda}), \tag{8}$$

where θ_k are learnable coefficients. Consequently, the filtering operation becomes:

$$y \approx \sum_{k=0}^{K} \theta_k T_k(\tilde{L})x. \tag{9}$$

This approximation localizes filtering to a K-hop neighborhood and reduces computational complexity. It enables

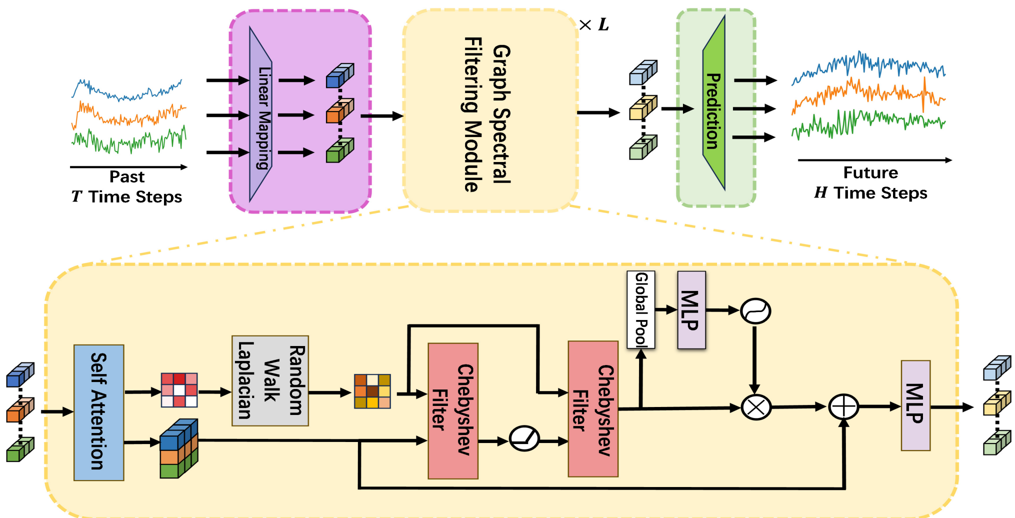

Fig. 1. **Overview of the proposed model for long-term multivariate time series forecasting.** The model consists of three main components: (1) **Temporal Embedding (Violet Block)**: Input multivariate time series $X \in \mathbb{R}^{B\times T\times N}$, where T is the historical window and N is the number of variables, is first permuted and projected through linear layers to extract temporal features for each variable independently, yielding embeddings in $\mathbb{R}^{B\times N\times D}$. (2) **Spectral Graph Filtering Module (Yellow Block)** (detailed in the lower panel): The temporal embeddings undergo spatial modeling through our core innovation, single head self-attention is firstly computed to generate a weighted adjacency matrix, which is then converted into a random walk Laplacian. Two-stage Chebyshev polynomial filtering is applied to learn frequency-aware spatial patterns. The filtered features are recalibrated via a Squeeze-and-Excitation [17] mechanism (Global Pool → MLP) and enhanced with a residual connection for stability. (3) **Prediction Layer (Green Block)**: The spatially-enhanced features are projected through a linear layer to produce H future time steps, resulting in the forecast $\hat{Y} \in \mathbb{R}^{B\times H\times N}$. This architecture enables adaptive learning of spatial patterns in the frequency domain, offering improved scalability and performance for high-dimensional multivariate forecasting compared to conventional attention-based methods.

efficient learning of spatial dependencies at multiple scales, capturing both localized interactions and global variable dynamics relevant for long-term forecasting.

IV. Methodology

A. Problem Formulation

Let $X \in \mathbb{R}^{B\times T\times N}$ denote the input multivariate time series, where B is the batch size, T is the historical time window, and N is the number of variables. The goal is to forecast the future H time steps for all N variables, producing the output $\hat{Y} \in \mathbb{R}^{B\times H\times N}$.

B. Model Architecture

The overall architecture of the proposed model consists of three main components: a temporal embedding layer, a stack of spectral graph filtering modules, and a prediction layer. First, the temporal embedding layer transforms the temporal sequence of each variable into a shared latent space, preserving variable-specific temporal periodic patterns. Next, the spectral graph filtering module captures inter-variable dependencies in a frequency-aware manner by computing attention-based Laplacians and applying multi-scale Chebyshev polynomial filters. Finally, the prediction layer maps the filtered features to future time steps via a linear projection. This modular design enables flexible and efficient long-term forecasting. An overview of the architecture is illustrated in Fig. 1.

1) Temporal Embedding Layer: Following the design of iTransformer [8], we first transform the temporal dynamics of each variable into a shared latent space. Given input $X \in \mathbb{R}^{B\times T\times N}$, we first permute the input to $\mathbb{R}^{B\times N\times T}$ and apply a linear projection across the temporal axis:

$$E_0 = \text{Linear}(X) \in \mathbb{R}^{B\times N\times d_{\text{model}}}, \tag{10}$$

where d_{model} is the embedding dimension.

2) Spectral Graph Filtering Module: To model spatial dependencies in a frequency-aware manner, we introduce a spectral graph filtering module composed of three stages: (1) learning pairwise interactions via attention mechanism, (2) constructing a Laplacian for spectral analysis, and (3) applying multi-scale Chebyshev filtering with feature recalibration.

a) Attention-Based Graph Construction.: We first learn a graph structure that captures spatial interactions of the input. From the temporal embedded variable representations $E_0 \in \mathbb{R}^{B\times N\times d_{\text{model}}}$, we compute spatial single-head raw attention scores [20] as follows:

$$W^{\text{raw}} = \frac{(E_0 W^Q)(E_0 W^K)^\top}{\sqrt{d_{\text{model}}}}, \tag{11}$$

where $W^Q, W^K \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ are learnable projection matrices. Applying row-wise softmax function then ensures normalized edge weights between each node:

$$W^{\text{attn}} = \text{softmax}(W^{\text{raw}}), \quad (12)$$

This attention matrix defines a weighted directed graph where $W^{\text{attn}}_{i,j} \in (0,1)$ represents the normalized influence strength from node i to node j. Unlike standard attention that directly transforms features, we use these weights to construct a graph Laplacian:

$$L = I - D^{-1}_{graph} W^{\text{attn}}, \quad (13)$$

where D_{graph} is the row-sum degree matrix of W^{attn}. This random walk normalized graph Laplacian enables spectral analysis while preserving the influence of edge weights.

b) Multi-Scale Spectral Filtering: We then approximate spectral filters using Chebyshev polynomials. First, we rescale the graph Laplacian by $\tilde{L} = \frac{2L}{\lambda_{\max}} - I$. We then perform two-stage Chebyshev filtering:

$$E^{(1)} = \sum_{k=0}^{K_1} \theta_k^{(1)} T_k(\tilde{L}) E_0, \quad (14)$$

$$E^{(2)} = \sum_{k=0}^{K_2} \theta_k^{(2)} T_k(\tilde{L}) E^{(1)}, \quad (15)$$

where $T_k(\cdot)$ denotes the k-th order Chebyshev polynomial and $\theta_k^{(\cdot)}$ are learnable filter coefficients. This sequential filtering captures both short-range and long-range variable interactions.

c) Feature Recalibration.: To enhance the relevance of learned features, we apply a Squeeze-and-Excitation [17] operation. First, a global average pooling over variables is computed as:

$$c = \frac{1}{N} \sum_{i=1}^{N} E_i^{(2)} \in \mathbb{R}^{d_{\text{model}}}, \quad (16)$$

it is then followed by a gating network:

$$s = \sigma(W_2 \, \text{ReLU}(W_1 c)) \in \mathbb{R}^{d_{\text{model}}}, \quad (17)$$

where $W_1 \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}/r}$, $W_2 \in \mathbb{R}^{d_{\text{model}}/r \times d_{\text{model}}}$ and r is a reduction ratio. The gated features are:

$$E^{\text{spec}} = E^{(2)} \odot s. \quad (18)$$

Then a MLP finalizes the feature recalibration:

$$E_{\text{out}} = \text{MLP}(E^{\text{spec}}). \quad (19)$$

3) Prediction Layer: The output of the spectral graph filtering module $H_{\text{out}} \in \mathbb{R}^{B \times N \times d_{\text{model}}}$ is passed to a linear prediction head, where the features are projected into the target horizon by:

$$\hat{Y} = \text{Linear}(E_{\text{spec}}) \in \mathbb{R}^{B \times N \times H}, \quad (20)$$

which is then permuted to match the output shape $\mathbb{R}^{B \times H \times N}$. This simple prediction head allows direct multi-step forecasting in a non-autoregressive fashion.

TABLE I
STATISTICS OF THE DATASETS USED IN OUR EXPERIMENTS.

Dataset	Electricity	Solar	Weather
# Variables	321	137	21
# Samples	18,317	36,601	36,792
# Frequency	60mins	10mins	10mins

V. EXPERIMENTS

A. Datasets

We evaluate our model on three widely-used real-world benchmarks for long-term MTSF: **Electricity** [4], **Solar** [21], and **Weather** [4]. These datasets span across diverse domains with varying temporal frequencies, spatial resolutions, and signal characteristics. Table I summarizes their key statistics.

- **Electricity**: Records hourly electricity consumption (in kWh) from 321 clients in the United States, spanning from 2012 to 2014.
- **Solar**: Contains 10-minute interval measurements of solar power generation from 137 synthetic photovoltaic (PV) sites in Alabama during the year 2006.
- **Weather**: Provides 10-minute resolution data for 21 meteorological variables (e.g., temperature, humidity, wind speed) across various locations during 2020.

B. Settings

Our model is implemented in PyTorch and trained on two NVIDIA RTX 3090 GPUs with an AMD EPYC 7301 CPU. We use the Adam optimizer along with an `ExponentialLR` learning rate scheduler. Early stopping is employed based on validation loss with a patience of 3 epochs. To maintain a fair comparison with other baselines, we follow prior works [4], [6], [8], [8] and split all datasets into training, validation, and testing sets using a fixed ratio of 7:1:2. The historical input length is fixed to $T = 96$, and the forecasting horizon H is set to one of the following values: $H \in \{96, 192, 336, 720\}$. We use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as evaluation metrics. All results are reported on the test set using the best model checkpoint determined by validation loss.

C. Baselines

We evaluate against nine well-known baselines introduced in Section II, representing two distinct spatial modeling paradigms for in long-term MTSF: (1) CI methods DLinear [5], RLinear [6], Autoformer [4], FEDformer [12], PatchTST [13], TiDE [14], and (2) spatial dependency modeling Crossformer [16], TimesNet [15], iTransformer [8]. These baselines span a wide range of architectural designs, offering a comprehensive evaluation of our spectral filtering approach against both spatial and non-spatial methods.

TABLE II
PERFORMANCE COMPARISON ON THE ELECTRICITY, SOLAR AND WEATHER DATASETS. THE INPUT LENGTH IS 96, WHILE THE PREDICTION LENGTHS ARE 96, 192, 336, 720. BEST RESULTS ARE IN RED BOLD, SECOND-BEST IN BLUE UNDERLINE.

Models		Proposed		iTransformer		RLinear		PatchTST		Crossformer		TiDE		TimesNet		DLinear		FEDformer		Autoformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Electricity	96	**0.144**	**0.237**	0.148	0.240	0.201	0.281	0.181	0.270	0.219	0.314	0.237	0.329	0.168	0.272	0.197	0.282	0.193	0.308	0.201	0.317
	192	**0.159**	**0.249**	0.162	0.253	0.201	0.283	0.188	0.274	0.231	0.322	0.236	0.330	0.184	0.289	0.196	0.285	0.201	0.315	0.222	0.334
	336	**0.177**	**0.269**	0.178	**0.269**	0.215	0.298	0.204	0.293	0.246	0.337	0.249	0.344	0.198	0.300	0.209	0.301	0.214	0.329	0.231	0.338
	720	**0.214**	**0.304**	0.225	0.317	0.257	0.331	0.246	0.324	0.280	0.363	0.284	0.373	0.220	0.320	0.245	0.333	0.246	0.355	0.254	0.361
	Avg	**0.173**	**0.264**	0.178	0.270	0.219	0.298	0.205	0.290	0.244	0.334	0.251	0.344	0.192	0.295	0.212	0.300	0.214	0.327	0.227	0.338
Solar	96	**0.199**	**0.235**	0.203	0.237	0.322	0.339	0.234	0.286	0.310	0.331	0.312	0.399	0.250	0.292	0.290	0.378	0.242	0.342	0.884	0.711
	192	**0.230**	**0.260**	0.233	0.261	0.359	0.356	0.267	0.310	0.734	0.725	0.339	0.416	0.296	0.318	0.320	0.398	0.285	0.380	0.834	0.692
	336	0.249	0.274	0.248	**0.273**	0.397	0.369	0.290	0.315	0.750	0.735	0.368	0.430	0.319	0.330	0.353	0.415	0.282	0.376	0.941	0.723
	720	0.252	0.280	**0.249**	**0.275**	0.397	0.356	0.289	0.317	0.769	0.765	0.370	0.425	0.338	0.337	0.356	0.413	0.357	0.427	0.882	0.717
	Avg	**0.233**	**0.262**	**0.233**	**0.262**	0.369	0.356	0.270	0.307	0.641	0.639	0.347	0.417	0.301	0.319	0.330	0.401	0.291	0.381	0.885	0.711
Weather	96	**0.170**	**0.208**	0.174	0.214	0.192	0.232	0.177	0.218	0.158	0.230	0.202	0.261	0.172	0.220	0.196	0.255	0.217	0.296	0.266	0.336
	192	**0.219**	**0.253**	0.221	0.254	0.240	0.271	0.225	0.259	0.206	0.277	0.242	0.298	0.219	0.261	0.237	0.296	0.276	0.336	0.307	0.367
	336	0.276	0.295	0.278	**0.296**	0.292	0.307	0.278	0.297	0.272	0.335	0.287	0.335	0.280	0.306	0.283	0.335	0.339	0.380	0.359	0.395
	720	0.356	0.348	0.358	**0.347**	0.364	0.353	0.354	0.348	0.398	0.418	0.351	0.386	0.365	0.359	**0.345**	0.381	0.403	0.428	0.419	0.428
	Avg	**0.255**	**0.276**	0.258	0.278	0.272	0.291	0.259	0.281	0.259	0.315	0.271	0.320	0.259	0.287	0.265	0.317	0.309	0.360	0.338	0.382

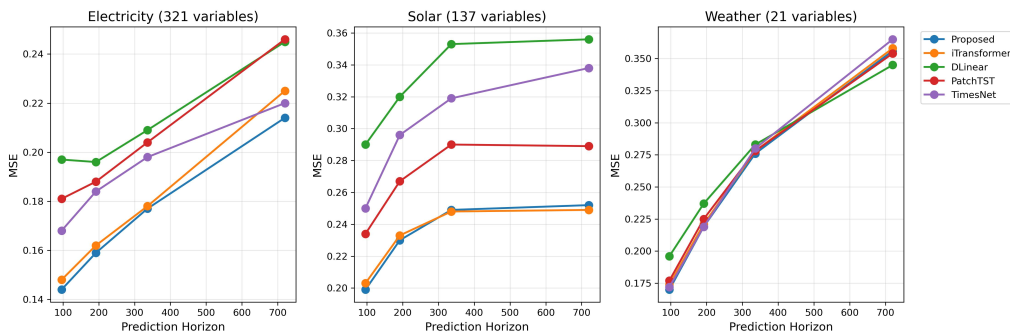

Fig. 2. Performance comparison across different prediction horizons. Each panel shows MSE values for five representative models on (a) Electricity (321 variables), (b) Solar (137 variables), and (c) Weather (21 variables) datasets. The proposed spectral filtering method (blue) consistently achieves the lowest error rates across all horizons on high-dimensional datasets, with particularly strong performance on Electricity.

D. Results and Analysis

Table II presents the performance comparison across the datasets: Electricity, Solar, and Weather. Each model is evaluated under four forecasting horizons $H \in \{96, 192, 336, 720\}$, representing increasing levels of difficulty. Fig. 2 visualizes these results, clearly showing the consistent superiority of our method across different prediction horizons. On the Electricity dataset, which contains 321 variables, our model consistently achieves the best performance across all horizons in both MAE and MSE. Compared to the closest competitor, iTransformer, our method reduces the average MSE by 2.8% and MAE by 2.2%. This strong performance highlights the effectiveness of our spectral filtering module in large network, where spatial dependencies are more complex. The superiority over iTransformer further supports our claim on the drawback of attention mechanism on a large number of variables. In the Solar dataset with 137 variables, our model performs comparably to iTransformer, with both models achieving identical average scores (0.233/0.262). Our model shows stronger performance at shorter horizons (96, 192), whereas iTransformer slightly outperforms at longer horizons (336, 720). This indicates that while both models can effectively model medium-scale spatial patterns, the benefits of spectral filtering are more prominent when short-term periodic patterns dominate. The drop in relative advantage at longer horizons may reflect an increase in noise or a weaker spatial signal across extended windows. The Weather dataset contains only 21 variables, and here, the gap between models narrows further. Interestingly, at $H = 720$, the channel-independent model DLinear achieves the best MSE, suggesting that spatial modeling offers diminishing returns when the variable count is small and long-term signal fidelity declines. This observation suggests that the spatial dependencies are either weak or irrelevant at such long horizons in low-dimensional datasets. Across datasets of increasing node count (Weather $<$ Solar $<$ Electricity), we observe that the advantage of our spectral graph filtering module scales accordingly. This scaling behavior is quantitatively demonstrated in Fig. 3, which reveals that the performance gap between our spectral filtering approach and attention-based methods grows monotonically the number of variables increases. This confirms that spectral filtering scales more gracefully with increasing variable count and captures spatial interactions more effectively than soft attention alone.

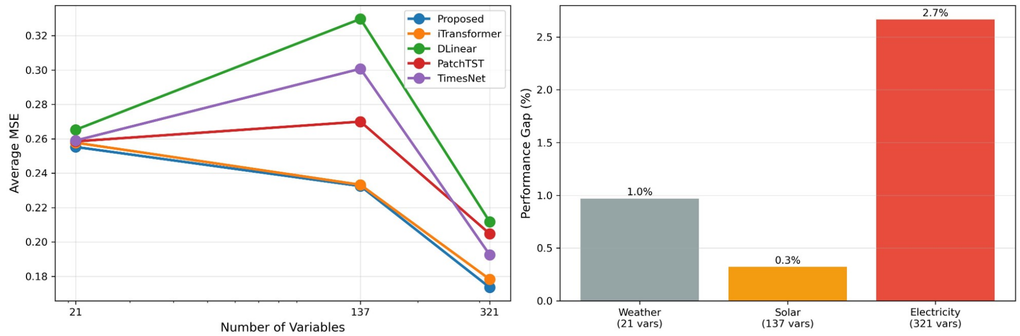

Fig. 3. Model performance over different variable dimensionalities across datasets. (a) Average MSE across all prediction horizons for key baseline methods, plotted against the number of variables in each dataset. (b) Relative performance gap between the proposed spectral filtering approach and iTransformer's attention-based spatial modeling. The gap increases from 137 variables to 321 variables, demonstrating that spectral filtering scales more effectively than direct attention mechanisms.

VI. Conclusion

In this work, we introduced a novel spectral graph filtering framework for long-term multivariate time series forecasting, integrating temporal encoding with frequency-aware spatial modeling. By transforming attention weights into graph Laplacians and applying multi-scale Chebyshev polynomial filters, our approach efficiently captures spatial dependencies, outperforming conventional attention-based methods in high-dimensional settings. Extensive experiments on Electricity, Solar, and Weather datasets demonstrate consistent improvements in MAE and RMSE. These results validate the effectiveness of spectral filtering for modeling complex multivariate dependencies and highlight its scalability for real-world applications such as energy management and climate forecasting. Future work will explore adaptive filter designs to enhance robustness and joint spatiotemporal modeling to better capture the dynamic evolution of multivariate signals.

References

[1] J.-S. Chou and D.-S. Tran, "Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders," *Energy*, vol. 165, pp. 709–726, 2018.

[2] Z. Karevan and J. A. Suykens, "Transductive lstm for time-series prediction: An application to weather forecasting," *Neural Networks*, vol. 125, pp. 1–9, 2020.

[3] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial time series forecasting with deep learning: A systematic literature review: 2005–2019," *Applied soft computing*, vol. 90, p. 106181, 2020.

[4] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[5] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 9, 2023, pp. 11 121–11 128.

[6] Z. Li, S. Qi, Y. Li, and Z. Xu, "Revisiting long-term time series forecasting: An investigation on linear mapping," *arXiv preprint arXiv:2305.10721*, 2023.

[7] L. Han, H.-J. Ye, and D.-C. Zhan, "The capacity and robustness trade-off: Revisiting the channel independent strategy for multivariate time series forecasting," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[8] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," *arXiv preprint arXiv:2310.06625*, 2023.

[9] H. Lee and S. Ko, "Testam: A time-enhanced spatio-temporal attention model with mixture of experts," *arXiv preprint arXiv:2403.02600*, 2024.

[10] S. Jin, H. Lee, C. Park, H. Chu, Y. Tae, J. Choo, and S. Ko, "A visual analytics system for improving attention-based traffic forecasting models," *IEEE transactions on visualization and computer graphics*, vol. 29, no. 1, pp. 1102–1112, 2022.

[11] X. Wang and M. Zhang, "How powerful are spectral graph neural networks," in *International conference on machine learning*. PMLR, 2022, pp. 23 341–23 362.

[12] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International conference on machine learning*. PMLR, 2022, pp. 27 268–27 286.

[13] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2022.

[14] A. Das, W. Kong, A. Leach, S. Mathur, R. Sen, and R. Yu, "Long-term forecasting with tide: Time-series dense encoder," *arXiv preprint arXiv:2304.08424*, 2023.

[15] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," *arXiv preprint arXiv:2210.02186*, 2022.

[16] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in *The eleventh international conference on learning representations*, 2023.

[17] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[18] F. Bauer, "Normalized graph laplacians for directed graphs," *Linear Algebra and its Applications*, vol. 436, no. 11, pp. 4193–4222, 2012.

[19] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in neural information processing systems*, vol. 29, 2016.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[21] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.