BucpTSF: Breaking the Uniform Computation Paradigm in Time-Series Forecasting

Hua Wang
School of Computer and Artificial Intelligence
Ludong University
Yantai 264025, China
hwa229@163.com

Jinghao Lu
School of Computer and Artificial Intelligence
Ludong University
Yantai 264025, China
ljh@m.ldu.edu.cn

Fan Zhang*
School of Computer Science and Technology
Shandong Technology and Business University
Yantai 264005, China
zhangfan@sdtbu.edu.cn

Abstract

Real-world time series typically exhibit non-uniform information density across temporal positions, frequency structures, and forecasting horizons: critical dynamics are concentrated in local segments, different frequency components carry different structural information, and different horizons require different modeling complexity. However, most existing methods still allocate computation approximately uniformly across these dimensions, leading to redundant computation in low-information regions and insufficient modeling of key patterns. To address this, we propose BucpTSF, an information-density-driven structured framework for time series forecasting. Specifically, BucpTSF reorganizes raw sequences into hierarchically heterogeneous multi-level token representations through information-density-aware temporal representation reorganization, captures key cross-token dependencies with low overhead via frequency-selective structured token interaction, and further introduces horizon-conditioned structured extrapolation to adaptively balance dynamic modeling and structural extrapolation bias for different horizons, thereby improving the accuracy and stability of long-term forecasting. Extensive experiments on real-world datasets demonstrate that BucpTSF achieves strong and competitive forecasting performance across diverse settings, supporting the effectiveness of structured non-uniform modeling for time-series forecasting.
Code Repository: https://github.com/SIXLJH/BucpTSF

1 Introduction

Time series forecasting is a fundamental task in machine learning with important applications in domains such as energy, transportation, climate, and finance Lu et al. [2025], Wang et al. [2024a]. Recent advances in Transformers, patch-based modeling, and multi-scale representation learning have substantially improved the modeling of complex temporal dependencies Kim et al. [2024]. However, most existing methods still implicitly follow a uniform computation allocation paradigm, applying roughly the same modeling complexity across temporal positions, frequency structures, and forecasting horizons Zhang et al. [2026], Xiao et al. [2026a]. This assumption mismatches

*Corresponding author.

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

the structure of real-world time series, whose information density often varies significantly across local temporal regions, frequency scales, and prediction horizons. This raises a key question: can computation be preferentially allocated to more informative temporal tokens, frequency patterns, and horizons, rather than uniformly over the entire sequence Wang et al. [2026a], Yao et al. [2026], Xiao et al. [2026b]?

Such non-uniformity is common in practice. In power and traffic forecasting, stable periodic trends often coexist with localized bursts, which are concentrated in a few critical periods; similarly, short-term prediction relies more on complex local dynamics, whereas long-term prediction is more about smooth structural extrapolation Zhang et al. [2024], Xiao and Hou [2026]. As shown in Fig. 1, temporal information density is typically highly concentrated, with a small number of local windows carrying most informative dynamics; meanwhile, spectral importance is not necessarily aligned with spectral energy, and low-amplitude high-frequency components may still contain key predictive cues Song et al. [2024], Wang et al. [2026b]. Therefore, imposing uniform modeling complexity over the entire sequence and all horizons leads to redundant computation in low-information regions and insufficient modeling of key dynamics Jiao et al. [2026], Wang et al. [2026c], Zhou et al. [2026].

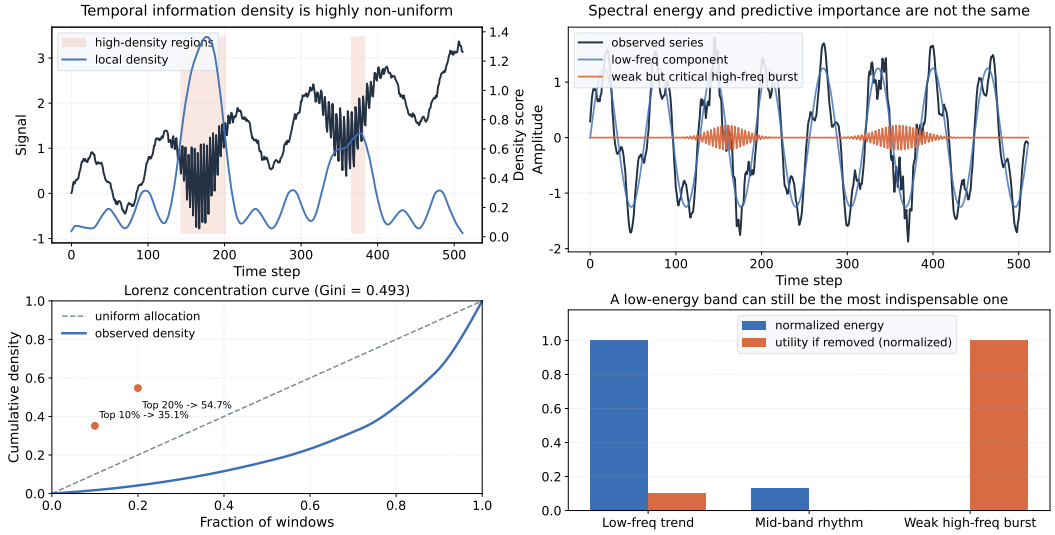

Figure 1: **Left:** Temporal information density is highly non-uniform, with a small fraction of local windows carrying most informative dynamics. **Right:** Spectral importance is not equivalent to spectral energy, and low-energy high-frequency components may still provide critical forecasting cues.

To address this issue, we propose BucpTSF (Breaking the Uniform Computation Paradigm for Time-Series Forecasting), an information-density-driven structured forecasting framework. BucpTSF realizes non-uniform computation allocation through three complementary stages. First, it reorganizes raw sequences into hierarchically heterogeneous multi-level token representations. Second, it performs frequency-selective token interaction in the spectral domain to capture structurally meaningful dynamic dependencies. Third, it uses horizon-conditioned structured extrapolation to adaptively balance expressive dynamic modeling and extrapolation bias, improving the stability and generalization of long-term forecasting.

In summary, our key contributions are as follows:

- We identify the implicit uniform computation allocation assumption in existing time-series forecasting models and argue for allocating modeling capacity according to temporal, spectral, and horizon-wise information density.
- We propose BucpTSF, which implements non-uniform computation allocation via temporal representation reorganization, frequency-selective structured token interaction, and horizon-conditioned structured extrapolation.

- Extensive experiments on more than ten public real-world datasets demonstrate that BucpTSF achieves strong and competitive forecasting performance across diverse long-term and short-term forecasting settings, while comprehensive ablation studies validate the complementary contributions of the proposed components.

2 Related Work

Prior work related to BucpTSF mainly falls into three lines: temporal representation reorganization, structured and frequency-aware token interaction, and horizon-aware forecasting. A more comprehensive discussion is provided in Appendix.

Temporal Representation Reorganization for Time-Series Forecasting Existing forecasting methods improve input representations through decomposition, rearrangement, and tokenization. Autoformer and TimesNet show that restructuring raw sequences into more structured spaces can ease pattern disentanglement and temporal modeling Wu et al. [2021, 2022]. PatchTST, iTransformer, and TimeXer further demonstrate that the definition and organization of tokens strongly affect forecasting performance and information integration efficiency Nie et al. [2022], Liu et al. [2023a], Wang et al. [2024a]. WaveToken goes one step further by treating tokenization itself as a core modeling component in a wavelet-based time–frequency space Masserano et al. [2025]. Different from these methods, BucpTSF reorganizes raw sequences into hierarchically heterogeneous tokens to explicitly support subsequent non-uniform computation allocation.

Structured and Frequency-Aware Token Interaction A second line of work focuses on making token interaction more structured and efficient. Informer and Pyraformer reduce the cost of dense attention through sparse or organized connectivity patterns Zhou et al. [2021], Liu et al. [2021]. ETSformer and FEDformer incorporate temporal priors or frequency-domain modeling into token interaction, while Fredformer highlights the frequency bias of Transformer-based forecasting models toward dominant low-frequency components Woo et al. [2022], Zhou et al. [2022], Piao et al. [2024]. Pathformer further introduces adaptive pathways for structured multi-scale interaction Chen et al. [2024]. In contrast, BucpTSF adopts a lightweight interaction operator with explicit spectral selectivity, aiming to concentrate modeling capacity on more informative cross-token dependencies.

Horizon-Aware Forecasting and Structured Extrapolation Recent long-horizon forecasting methods also suggest that different forecasting steps may not require the same prediction complexity. DLinear and TSMixer show that strong forecasting performance can arise from simple prediction heads or lightweight backbones when the inductive bias is appropriate Zeng et al. [2023], Ekambaram et al. [2023]. FITS, MixLinear, and OLinear further support the value of compact and structured extrapolation, especially for long-range forecasting Xu et al. [2023], Ma et al. [2024], Yue et al. [2025]. However, these methods typically apply a largely uniform prediction form across all horizons. BucpTSF instead explicitly introduces horizon-conditioned structured extrapolation, allowing the balance between dynamic modeling and structural extrapolation to vary with the forecasting horizon.

3 Method

3.1 Problem Formulation and Method Overview

Given a multivariate time series $\mathbf{X} = \{x_1, x_2, \ldots, x_L\} \in \mathbb{R}^{L \times C}$ of length L with C variables, where $x_t \in \mathbb{R}^C$ denotes the multivariate observation at time step t, the goal of time-series forecasting is to predict the values of the next H time steps from the historical observations $\mathbf{X}$, yielding the forecast sequence $\hat{\mathbf{Y}} = \{\hat{x}_{L+1}, \hat{x}_{L+2}, \ldots, \hat{x}_{L+H}\} \in \mathbb{R}^{H \times C}$. Formally, this corresponds to learning a mapping $f_\theta : \mathbb{R}^{L \times C} \to \mathbb{R}^{H \times C}$ such that $\hat{\mathbf{Y}} = f_\theta(\mathbf{X})$, where θ denotes the model parameters.

Most existing forecasting methods use approximately uniform modeling complexity across temporal positions, frequency structures, and forecasting horizons, whereas real-world time series are typically highly non-uniform along these dimensions. This motivates allocating representation, interaction, and prediction capacity according to the information density of different structural components.

As illustrated in Fig. 2, BucpTSF is an information-density-driven structured forecasting framework with three stages: **(1) Information-Density-Aware Temporal Representation Reorganization**,

which converts raw sequences into hierarchically heterogeneous tokens through multi-resolution decomposition, patching, and tokenization; **(2) Frequency-Selective Structured Token Interaction**, which emphasizes informative cross-token dependencies via frequency-selective interactions; and **(3) Horizon-Conditioned Structured Extrapolation**, which adaptively balances a high-capacity dynamic branch and a low-complexity structured extrapolation branch across forecasting horizons.

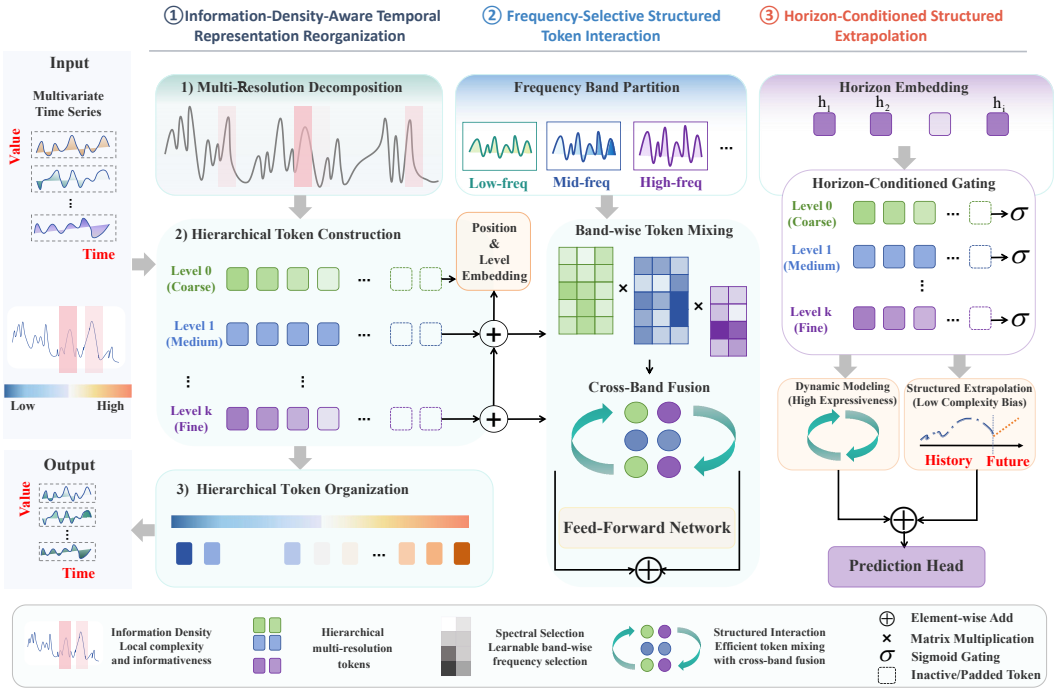

Figure 2: Overview of BucpTSF, which allocates computation non-uniformly through representation reorganization, structured token interaction, and horizon-conditioned extrapolation.

3.2 Information-Density-Driven Computation Allocation Principle

As illustrated in Fig. 3, forecasting-relevant structure varies across temporal locations, spectral modes, and prediction horizons. This motivates a design preference to allocate effective modeling capacity according to structural informativeness:

$$C(t, s, h) \propto I(t, s, h), \tag{1}$$

where t, s, and h index temporal position, frequency scale, and horizon, respectively. Equation (1) is a *conceptual architectural principle*, not a fitted proportionality constraint: $I(t, s, h)$ is neither directly observed nor explicitly supervised, and no uniquely identifiable density ground truth is assumed.

BucpTSF implements this preference with three differentiable structural proxies: RR reorganizes multiresolution temporal tokens, FSI learns frequency-dependent interaction strengths over those tokens, and HSE changes the relative contribution of nonlinear dynamics and structured extrapolation across future steps. Accordingly, $C(t, s, h)$ refers to *effective representational capacity, interaction strength, and prediction bias*, not per-sample floating-point operations. For fixed input and prediction lengths, BucpTSF uses a fixed computational graph without token pruning, early exit, or conditional execution; its FLOPs do not adapt to individual samples. The alignment of RR/FSI responses with independently measured predictive utility is examined in Appendix B.9.

3.3 Information-Density-Aware Temporal Representation Reorganization

To expose the structural heterogeneity entangled in the raw temporal domain, we first map the input sequence into a hierarchical representation space via a reversible multiresolution transform.

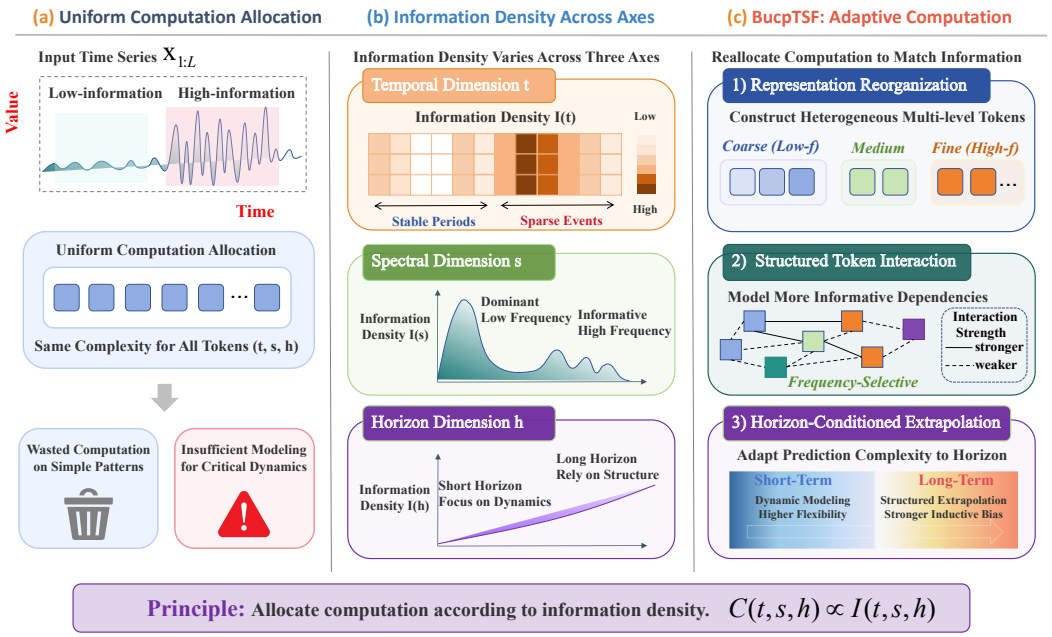

Figure 3: Motivation and principle of BucpTSF. (a) Uniform computation allocation applies similar complexity to all structures, causing wasted computation and weak modeling of critical dynamics. (b) Information density in real-world time series is non-uniform across temporal, spectral, and horizon dimensions. (c) BucpTSF reallocates model capacity through representation reorganization, structured token interaction, and horizon-conditioned extrapolation.

For the input sequence $\mathbf{X} \in \mathbb{R}^{L\times C}$, we define a reversible multiresolution transform

$$\mathcal{T} : \mathbb{R}^{L\times C} \to \{\mathbf{Z}_0, \mathbf{Z}_1, \ldots, \mathbf{Z}_K\}, \tag{2}$$

which yields

$$(\mathbf{Z}_0, \mathbf{Z}_1, \ldots, \mathbf{Z}_K) = \mathcal{T}(\mathbf{X}). \tag{3}$$

Here, $\mathbf{Z}_0 \in \mathbb{R}^{L_0\times C}$ denotes the low-frequency approximation component that captures the dominant long-range trend, while $\mathbf{Z}_k \in \mathbb{R}^{L_k\times C}$ $(k = 1, \ldots, K)$ denotes the high-frequency detail component at the k-th level, capturing local fluctuations and transient variations at different scales. This decomposition separates the raw sequence into subspaces with different structural complexity.

For unified modeling, we further apply patchification to each resolution component. For the k-th component $\mathbf{Z}_k$, the i-th patch is defined as

$$\mathbf{P}_{k,i} = \mathbf{Z}_k[i : i + p, :] \in \mathbb{R}^{p\times C}, \tag{4}$$

where p denotes the patch length and i denotes the starting temporal index. We then project each patch into a shared latent space:

$$\mathbf{E}_{k,i} = \mathrm{vec}(\mathbf{P}_{k,i})W_e + b_e, \tag{5}$$

where $W_e \in \mathbb{R}^{(pC)\times d}$ is the patch embedding matrix, $b_e \in \mathbb{R}^d$ is the bias term, and d denotes the token embedding dimension.

The resulting token sequence at the k-th resolution is

$$\mathbf{E}_k = \{\mathbf{E}_{k,1}, \mathbf{E}_{k,2}, \ldots, \mathbf{E}_{k,N_k}\} \in \mathbb{R}^{N_k\times d}, \tag{6}$$

where N_k denotes the number of tokens at the k-th resolution. The entire input sequence is therefore reorganized into a hierarchical token set

$$\mathcal{E} = \{\mathbf{E}_0, \mathbf{E}_1, \ldots, \mathbf{E}_K\}. \tag{7}$$

This reorganization disentangles low- and high-frequency structures and provides a unified multi-scale token interface for subsequent structured interaction and prediction.

3.4 Frequency-Selective Structured Token Interaction

After representation reorganization, we obtain hierarchical token sequences at different resolutions, $\mathcal{E} = \{\mathbf{E}_0, \ldots, \mathbf{E}_K\}$. Instead of performing dense pairwise interaction among all tokens, we model their dependencies through a frequency-selective operator, since relations among patch-level tokens often exhibit structured periodic and modulation patterns.

For a token sequence at an arbitrary resolution level,

$$\mathbf{E} \in \mathbb{R}^{N \times d}, \tag{8}$$

where N denotes the number of tokens at that level and d denotes the token embedding dimension, we first apply a discrete Fourier transform along the token dimension to obtain the frequency representation

$$\mathbf{F} = \mathcal{F}(\mathbf{E}), \tag{9}$$

where $\mathcal{F}(\cdot)$ denotes the Fourier transform operator and $\mathbf{F} \in \mathbb{C}^{N \times d}$ is the complex-domain spectral representation.

In the frequency domain, we introduce a learnable complex filter

$$H \in \mathbb{C}^{N \times 1}, \tag{10}$$

which is broadcast along the embedding dimension, and modulate the spectrum via

$$\mathbf{F}' = H \odot \mathbf{F}, \tag{11}$$

where $\odot$ denotes element-wise multiplication. This filter emphasizes informative frequency modes and suppresses weak or noisy ones, thereby focusing interaction capacity on structurally meaningful inter-token dependencies.

We then map the modulated spectrum back to the token space and take its real part:

$$\mathbf{E}' = \mathrm{Re}\big(\mathcal{F}^{-1}(\mathbf{F}')\big), \tag{12}$$

where $\mathcal{F}^{-1}(\cdot)$ denotes the inverse Fourier transform. To preserve the original token information and improve training stability, we further adopt a residual update

$$\tilde{\mathbf{E}} = \mathbf{E} + \alpha(\mathbf{E}' - \mathbf{E}), \tag{13}$$

where α is a learnable scalar coefficient controlling the strength of frequency interaction.

Unlike Fourier processing applied directly to raw series, FSI modulates modes along the token axis *after* multiresolution reorganization and restores the resulting token representation through an inverse transform and residual update. The Fourier transform and complex filtering are established operators; the contribution here is their role in structured inter-token interaction within the coordinated RR–FSI–HSE framework, rather than a new transform or a sample-dependent hard-selection rule.

3.5 Horizon-Conditioned Structured Extrapolation

After representation reorganization and structured token interaction, the model already has strong temporal representations. However, different forecasting horizons require different prediction complexity: short horizons depend more on complex local dynamics, whereas long horizons benefit more from smooth structural extrapolation. We therefore introduce a horizon-conditioned structured extrapolation mechanism to adaptively balance these two predictive biases.

Let the backbone output after the above modules be

$$\mathbf{Y}_b \in \mathbb{R}^{H \times C}, \tag{14}$$

where $\mathbf{Y}_b$ denotes the nonlinear prediction for the next H time steps. Owing to its strong expressiveness, this branch is better suited for modeling short-term complex dynamics and nonlinear multiscale patterns.

We additionally introduce a low-rank structured extrapolation branch:

$$\mathbf{Y}_l = W_u W_d \mathbf{X}, \tag{15}$$

where $W_d \in \mathbb{R}^{r\times L}$ is a down-projection matrix, $W_u \in \mathbb{R}^{H\times r}$ is an up-projection matrix, and r denotes the low-rank dimension with $r \ll \min(L, H)$. Accordingly, $\mathbf{Y}_l \in \mathbb{R}^{H\times C}$ is the prediction produced by the structured extrapolation branch. Compared with the nonlinear backbone, this branch favors smoother forecasts with stronger inductive bias and is therefore more suitable for distant horizons.

To adapt the predictive bias across horizons, we define a monotonically increasing gating function

$$\beta(h) = 1 - \exp(-\lambda h), \qquad h \in \{1, 2, \ldots, H\}, \tag{16}$$

where $\lambda \geq 0$ is a learnable nonnegative parameter. Since $\beta(h)$ increases with h, the model relies more on the backbone at near horizons and gradually increases the contribution of the structured extrapolation branch at far horizons.[2]

The parameter λ is learned for each dataset–horizon configuration, whereas $\beta(h)$ is monotone only *within* that fixed configuration. This structural prior does not imply monotonic error reductions as the prediction horizon grows, nor is a monotone gate necessarily optimal for every process. Appendix B.10 reports the learned gate behavior for ETTh1 and Traffic.

The final prediction is

$$\hat{\mathbf{Y}} = (\mathbf{1} - \boldsymbol{\beta}) \odot \mathbf{Y}_b + \boldsymbol{\beta} \odot \mathbf{Y}_l, \tag{17}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{H\times 1}$ is the horizon-wise gating vector composed of $\beta(h)$ and broadcast along the variable dimension, and $\odot$ denotes element-wise scaling. This fusion yields horizon-varying predictive complexity and completes non-uniform computation allocation at the prediction level.

4 Experiments

We evaluate **BucpTSF** on multiple benchmarks covering long-term forecasting and short-term traffic forecasting, with diverse dynamic complexity and variable scales. We further conduct ablation studies and targeted analyses to examine the roles of representation reorganization, structured token interaction, and horizon-conditioned prediction. For representative datasets, we also evaluate BucpTSF under three random seeds, with standard deviations no larger than 0.002, as reported in Appendix B.8. [3]

4.1 Experimental Setup and Baselines

Datasets. We conduct experiments on three representative settings: (1) standard long-term forecasting benchmarks, including ETT, Weather, Electricity, and Traffic; (2) low-dimensional datasets with complex dependency structures; and (3) short-term traffic forecasting benchmarks, including the PeMS series.

Baselines. For long-term forecasting, we compare BucpTSF with recent strong baselines, including TimeBase Huang et al. [2025], AMD Hu et al. [2025], SparseTSF Lin et al. [2024], TimeMixer Wang et al. [2024b], iTransformer Liu et al. [2023b], DeformableTST Luo and Wang [2024], and PatchTST Nie et al. [2022]. For short-term traffic forecasting, we follow the standard evaluation protocol and compare against representative methods. We adopt mean squared error (MSE) and mean absolute error (MAE) as the main evaluation metrics.

Result provenance. For Table 1, the TimeBase and SparseTSF entries are published comparisons reported by TimeBase, AMD is taken from its official results, and the other baseline entries follow the TimeBridge Liu et al. [2024] comparison. Baselines in Table 2 are published comparisons reported by TimeBase. For the PEMS results in Table 3, we independently reproduced all compared models using the same preprocessing, splits, look-back/prediction lengths, metrics, and evaluation pipeline. Published baseline values should not be interpreted as runs reproduced in our environment.

[2] We compare alternative horizon gating functions, including uniform, linear, and MLP-based gates, in Appendix B.5.

[3] The selection of hyperparameters and the details of all datasets are presented in the appendix B.

Table 1: Long-term forecasting performance on traditional datasets, averaged over prediction lengths $H \in \{96, 192, 336, 720\}$. The best results are highlighted in bold.

Models	H	Ours		2025(l_{weight})		2024(l_{weight})		2025		2023–2024							
		BucpTSF		TimeBase		SparseTSF		AMD		iTransformer		DeformableTST		TimeMixer		PatchTST	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	Avg	**0.343**	**0.373**	0.357	0.381	0.362	0.384	0.347	0.376	0.362	0.391	0.348	0.383	0.356	0.380	0.353	0.376
ETTm2	Avg	**0.250**	**0.311**	0.251	0.314	0.253	0.317	0.254	0.316	0.269	0.329	0.257	0.319	0.257	0.318	0.256	0.317
ETTh1	Avg	**0.384**	**0.414**	0.396	0.415	0.407	0.419	0.407	0.424	0.439	0.448	0.404	0.423	0.427	0.441	0.413	0.434
ETTh2	Avg	0.328	0.381	0.347	0.398	0.344	0.387	0.351	0.392	0.374	0.406	0.328	**0.377**	0.349	0.397	**0.324**	0.381
WTH	Avg	**0.219**	**0.254**	**0.219**	0.263	0.244	0.286	0.223	0.263	0.233	0.271	0.222	0.262	0.226	0.264	0.226	0.264
Traffic	Avg	**0.385**	0.265	0.418	0.279	0.414	0.280	0.394	0.272	0.397	0.282	0.391	0.278	0.409	0.279	0.391	**0.264**
ECL	Avg	**0.157**	**0.250**	0.167	0.258	0.168	0.262	0.159	0.254	0.164	0.261	0.161	0.261	0.185	0.284	0.159	0.253

Table 2: The long-term prediction performance of BucpTSF on low-dimensional datasets

Models	H	BucpTSF(ours)		TimeBase		SparseTSF		TimeMixer		FITS		iTransformer		DLinear		PatchTST	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
AQshunyi	Avg	**0.669**	**0.490**	0.675	0.506	0.760	0.546	0.736	0.536	0.763	0.548	0.723	0.520	0.757	0.572	0.691	0.503
AQWan	Avg	**0.771**	**0.481**	0.779	0.499	0.826	0.526	0.822	0.521	0.813	0.519	0.843	0.538	0.883	0.560	0.810	0.513
CzeLan	Avg	**0.207**	**0.247**	0.225	0.262	0.241	0.292	0.234	0.282	0.250	0.304	0.245	0.297	0.242	0.290	0.229	0.276
Temp	Avg	**0.181**	**0.333**	0.187	0.338	0.295	0.413	0.302	0.423	0.315	0.442	0.302	0.423	0.292	0.418	0.291	0.406
Wind	Avg	**0.933**	**0.695**	0.940	0.709	1.014	0.737	1.044	0.758	1.023	0.746	1.025	0.748	1.018	0.739	0.968	0.703

4.2 Long-Term Time-Series Forecasting

As shown in Table 1 and Table 2, BucpTSF achieves strong and competitive forecasting performance across diverse long-term forecasting benchmarks, attaining the best results on multiple datasets while remaining comparable to the strongest baselines on others. These results suggest that coordinating heterogeneous temporal representations, frequency-selective interactions, and horizon-conditioned extrapolation can effectively capture diverse temporal dynamics. The favorable performance on both standard benchmarks and low-dimensional datasets further indicates that the proposed framework is applicable across different variable scales and dependency structures. Overall, the results provide empirical support for structured non-uniform modeling as an effective approach to long-term time-series forecasting.

4.3 Short-Term Time-Series Forecasting

As shown in Table 3, BucpTSF achieves the best or highly competitive results on short-term forecasting tasks across multiple datasets. On traffic benchmarks such as PEMS, which feature strong high-frequency dynamics, complex short-range dependencies, and pronounced cross-variable coupling, BucpTSF more effectively captures critical local patterns by allocating modeling capacity according to information density across temporal regions, frequency structures, and forecasting horizons. These results further demonstrate that the proposed non-uniform computation allocation mechanism is effective not only for long-horizon extrapolation, but also for multivariate short-term forecasting with complex local dynamics.

Table 3: Short-term forecasting performance on the PEMS datasets, averaged over prediction lengths $H \in \{12, 24, 48, 96\}$. The best results are highlighted in bold.

Models	H	BucpTSF		Twins*		itrans*		RLinear		PatchTST		Cross*		TiDE		TimesNet	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS03	Avg	**0.090**	**0.195**	0.109	0.219	0.116	0.226	0.495	0.472	0.180	0.304	0.169	0.272	0.376	0.379	0.159	0.262
PEMS04	Avg	**0.097**	**0.203**	0.111	0.219	0.121	0.232	0.526	0.491	0.195	0.307	0.209	0.317	0.353	0.420	0.130	0.241
PEMS07	Avg	**0.089**	**0.192**	0.094	0.196	0.100	0.204	0.505	0.478	0.211	0.303	0.235	0.315	0.380	0.440	0.125	0.226
PEMS08	Avg	**0.122**	**0.198**	0.133	0.222	0.151	0.234	0.529	0.487	0.280	0.321	0.268	0.307	0.441	0.464	0.193	0.271

* indicates a Transformer-based model.

4.4 Ablation Study

As shown in Table 4 and Fig. 4, RR, FSI, and HSE each improve over Base, their pairwise combinations yield further gains, and the complete model performs best on all four ablation datasets. The uniform variants remain weaker than the full model, supporting the benefit of heterogeneous representation, interaction, and horizon-specific fusion beyond module stacking. Fig. 4(C) shows positive but non-monotonic gains across all 16 evaluated dataset–horizon settings. Relative to Base,

BucpTSF reduces average MSE by 16.3%, 14.9%, 9.6%, and 46.4% on ETTh1, ETTm1, AQWan, and PEMS03, respectively. Fig. 4(D) illustrates the accuracy–complexity trade-off; the associated complexity index should not be confused with input-dependent FLOPs.

Mechanistic and resource checks. To evaluate whether the structural proxies respond to useful forecasting evidence, we measure the loss increase when temporal regions or spectral modes are occluded, and correlate this utility with the corresponding model responses. Across six settings, the mean Spearman correlations are 0.497 (temporal) and 0.549 (spectral); these are alignment diagnostics, not estimates of a supervised information-density ground truth (Appendix B.9). Matched inference measurements and learned gate trajectories are reported in Appendices B.11 and B.10, respectively.

Table 4: Ablation results averaged over forecasting horizons (MSE, lower is better).

Dataset	Base	+RR	+FSI	+HSE	+RR+FSI	+RR+HSE	+FSI+HSE	U-RR	U-FSI	U-HSE	BucpTSF
ETTh1	0.459	0.418	0.415	0.425	0.399	0.406	0.399	0.394	0.398	0.390	**0.384**
ETTm1	0.403	0.372	0.367	0.380	0.355	0.361	0.356	0.351	0.354	0.348	**0.343**
AQWan	0.853	0.804	0.798	0.808	0.784	0.794	0.790	0.784	0.782	0.778	**0.771**
PEMS03	0.168	0.135	0.131	0.139	0.118	0.126	0.122	0.116	0.119	0.112	**0.090**

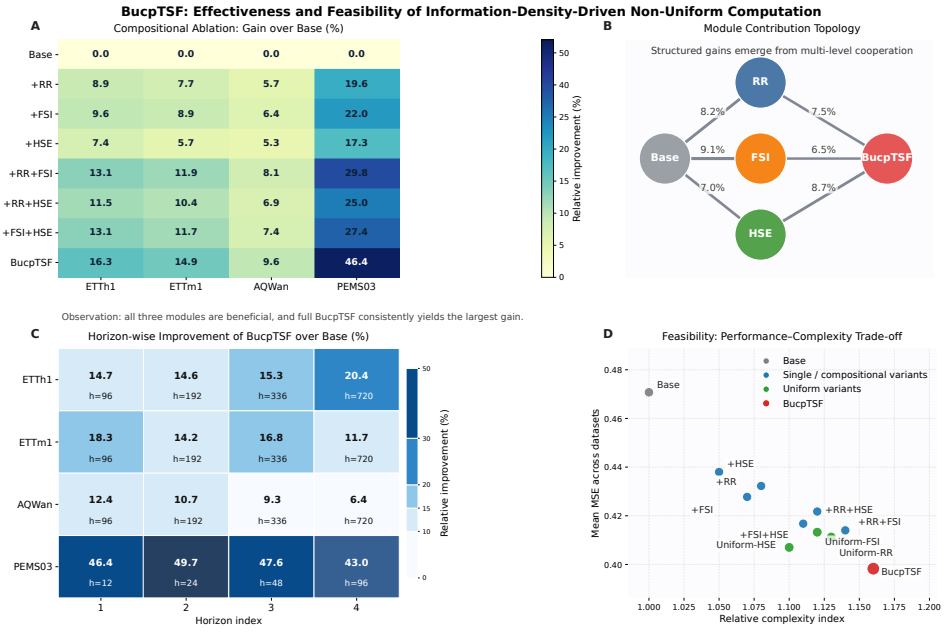

Figure 4: Effectiveness and feasibility analysis of BucpTSF. (A) Relative gains of compositional ablations over the Base backbone across datasets. (B) Topological visualization of module contributions from Base to the full BucpTSF. (C) Horizon-wise relative improvements of BucpTSF over Base, showing consistent gains across different forecasting lengths. (D) Performance–complexity trade-off, indicating that BucpTSF achieves a more favorable Pareto position than both compositional and uniform variants.

4.5 Mechanistic Validation and Computational Efficiency

Predictive-utility alignment. To test whether the structural responses capture forecasting-relevant information, we occlude a temporal region or spectral mode k and measure the corresponding increase in prediction loss:

$$U_k = \mathcal{L}(\hat{\mathbf{Y}}_{-k}, \mathbf{Y}) - \mathcal{L}(\hat{\mathbf{Y}}, \mathbf{Y}), \tag{18}$$

where $\hat{\mathbf{Y}}_{-k}$ denotes the prediction with component k masked. We compute Spearman's rank correlation between U_k and the temporal representation response or learned spectral-filter magnitude.

Table 5: Mechanistic validation and matched inference efficiency of BucpTSF. (a) Spearman correlations between occlusion utility and model responses; (b) resource measurements on ETTh1 ($L = 512$, $H = 720$, batch size 1).

(a) Predictive-utility alignment

Dataset	H	Temporal ρ	Spectral ρ
AQWan	96	0.397	0.433
AQWan	192	0.686	0.729
ETTh1	96	0.748	0.638
ETTh1	192	0.367	0.476
PEMS03	12	0.399	0.595
PEMS03	24	0.385	0.422
Mean	–	**0.497**	**0.549**

(b) Matched inference efficiency

Metric	PatchTST	BucpTSF
Parameters (M)	6.30	**3.91**
MACs (M)	225	**182**
Peak memory (MB)	672	**79**
Latency (ms)	10.0	**3.8**

Each setting uses 128 test samples with seed 42. Table 5 (left) reports positive correlations for all six settings, with mean coefficients of 0.497 (temporal) and 0.549 (spectral). These results support the alignment of RR/FSI responses with predictive utility; they do not establish an explicitly estimated information-density function.

Matched inference resources. Table 5 (right) compares BucpTSF with PatchTST under the same ETTh1 setting ($L = 512$, $H = 720$, batch size 1, same GPU). BucpTSF uses fewer parameters and MACs, lower peak inference memory, and shorter latency in this setting. This reflects the practical efficiency of its fixed computational graph, not sample-dependent FLOPs allocation.

5 Limitations

Although BucpTSF achieves strong empirical performance across diverse forecasting benchmarks, it introduces additional architectural components compared with minimal forecasting backbones. Moreover, our evaluation mainly focuses on standard public benchmarks and selected low-dimensional datasets, while broader deployment scenarios with distribution shift, missing values, or streaming constraints remain to be further explored.

6 Conclusion and Future Work

We proposed BucpTSF, an information-density-driven structured framework for time-series forecasting, which implements non-uniform modeling across representation, interaction, and prediction through temporal representation reorganization, frequency-selective interaction, and horizon-conditioned structured extrapolation. Extensive experiments on long-term and short-term forecasting benchmarks demonstrate that BucpTSF achieves strong and competitive forecasting performance across diverse settings, supporting the effectiveness of structured non-uniform modeling for time-series forecasting.

Future work may proceed in three directions: developing more explicit and interpretable information-density estimation mechanisms, extending the framework to more challenging settings such as missing values, irregular sampling, distribution shift, and exogenous variables, and improving efficiency for ultra-long sequences and extremely high-dimensional inputs to achieve a better performance–complexity trade-off. We hope this work provides a useful perspective on adaptive computation allocation in time-series forecasting.

Acknowledgment

This work was supported in part by the following: the National Natural Science Foundation of China under Grant Nos. U24A20328, U24A20219, 62272281, the Natural Science Foundation of Shandong Province under Grant No. ZR2026MS1059, the Youth Innovation Technology Project of Higher School in Shandong Province under Grant No. 2023KJ212, and the Yantai Natural Science Foundation under Grant No. 2024JCYJ034.

Competing Interests. The authors declare no competing interests.

References

Jinghao Lu, Fan Zhang, Xiaofeng Zhang, Yujuan Sun, and Hua Wang. Mcnr: Multiscale feature-based latent data component extraction linear regression model. *Expert Systems with Applications*, 292: 128634, 2025.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. Timexer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 37:469–498, 2024a.

Dongbin Kim, Jinseong Park, Jaewook Lee, and Hoki Kim. Are self-attentions effective for time series forecasting? *Advances in Neural Information Processing Systems*, 37:114180–114209, 2024.

Jiayu Zhang, Chuangxin Zhao, Canran Xiao, Ruibo Duan, Wenyi Mo, Haoyu Gao, and Wenshuo Wang. Pi-cca: Prompt-invariant cca certificates for replay-free continual multimodal learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

Canran Xiao, Liaoyuan Fan, Yanbin Li, Jing Tang, and Peilai Yu. Path matters: Unveiling geometric implicit bias via curvature-aware sparse view optimization. In *International Conference on Learning Representations*, volume 2026, pages 125264–125292, 2026a.

Hua Wang, Jinghao Lu, and Fan Zhang. Idealtsf: Can non-ideal data contribute to enhancing the performance of time series forecasting models? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 26224–26232, 2026a.

Zhen Yao, Bo Lang, Zhihao Zheng, Haoyan Xu, Zhengtao Yao, Zhikang Dong, and Mooi Choo Chuah. From semantic to reasoing: A survey of language-guided segmentation in the multimodal era. 2026.

Canran Xiao, Jiabao Dou, Zhiming Lin, Zong Ke, and Liwei Hou. From points to coalitions: Hierarchical contrastive shapley values for prioritizing data samples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 15995–16003, 2026b.

Jiawen Zhang, Shun Zheng, Xumeng Wen, Xiaofang Zhou, Jiang Bian, and Jia Li. Elastst: Towards robust varied-horizon forecasting with elastic time-series transformer. *Advances in Neural Information Processing Systems*, 37:119174–119197, 2024.

Canran Xiao and Liwei Hou. Prototype-aligned federated soft-prompts for continual web personalization. In *Proceedings of the ACM Web Conference 2026*, pages 6743–6754, 2026.

Yuchen Song, Min Zhang, Xiaotian Jiang, Fan Zhang, Cheng Ju, Shanguo Huang, Alan Pak Tao Lau, and Danshi Wang. Srs-net: a universal framework for solving stimulated raman scattering in nonlinear fiber-optic systems by physics-informed deep learning. *Communications engineering*, 3 (1):109, 2024.

Hua Wang, Xianhao Jiao, and Fan Zhang. PESD-TSF: A period-aware and explicit structured decomposition framework for long-term time series forecasting. In Tong Zhang, Miroslav Dudik, Martin Jaggi, Alekh Agarwal, Sharon Li, Dale Schuurmans, Jerry Zhu, Felix Berkenkamp, Hanze Dong, and Alberto Bietti, editors, *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 127230–127252. PMLR, 06–11 Jul 2026b. URL `https://proceedings.mlr.press/v306/wang26cs.html`.

Xianhao Jiao, Fan Zhang, and Hua Wang. Timescalex: Multiscale decomposition and fusion learning for long-term time series forecasting. *Applied Soft Computing*, 201:115581, 2026.

Hua Wang, Jinghao Lu, and Fan Zhang. Eeo-tfv: Escape-explore optimizer for web-scale time-series forecasting and vision analysis. In *Proceedings of the ACM Web Conference 2026*, pages 7271–7282, 2026c.

Heng Zhou, Jing Tang, Jusheng Zhang, Yanshu Li, Canran Xiao, Liwei Hou, Zong Ke, and Jiawei Yao. Comem: Compositional concept-graph memory for vision–language adaptation. In *International Conference on Learning Representations*, volume 2026, pages 20009–20032, 2026.

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

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2022.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2022.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2023a.

Luca Masserano, Abdul Fatir Ansari, Boran Han, Xiyuan Zhang, Christos Faloutsos, Michael W Mahoney, Andrew Gordon Wilson, Youngsuk Park, Syama Sundar Rangapuram, Danielle C Maddix, et al. Enhancing foundation models for time series forecasting via wavelet-based tokenization. In *International Conference on Machine Learning*, pages 43248–43275. PMLR, 2025.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 11106–11115, 2021.

Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*, 2021.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Etsformer: Exponential smoothing transformers for time-series forecasting. *arXiv preprint arXiv:2202.01381*, 2022.

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

Xihao Piao, Zheng Chen, Taichi Murayama, Yasuko Matsubara, and Yasushi Sakurai. Fredformer: Frequency debiased transformer for time series forecasting. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 2400–2410, 2024.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *ICLR 2024: The Twelfth International Conference on Learning Representations.*, 2024.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 11121–11128, 2023.

Vijay Ekambaram, Arindam Jati, Nam Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. Tsmixer: Lightweight mlp-mixer model for multivariate time series forecasting. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 459–469, 2023.

Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with $10k$ parameters. In *The Twelfth International Conference on Learning Representations*, 2023.

Aitian Ma, Dongsheng Luo, and Mo Sha. Mixlinear: Extreme low resource multivariate time series forecasting with 0.1 k parameters. *arXiv preprint arXiv:2410.02081*, 2024.

Wenzhen Yue, Yong Liu, Hao Wang, Haoxuan Li, Xianghua Ying, Ruohao Guo, Bowei Xing, and Ji Shi. Olinear: A linear model for time series forecasting in orthogonally transformed domain. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Qihe Huang, Zhengyang Zhou, Kuo Yang, Zhongchao Yi, Xu Wang, and Yang Wang. Timebase: The power of minimalism in efficient long-term time series forecasting. In *Forty-second International Conference on Machine Learning*, 2025.

Yifan Hu, Peiyuan Liu, Peng Zhu, Dawei Cheng, and Tao Dai. Adaptive multi-scale decomposition framework for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 17359–17367, 2025.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and Junjie Yang. Sparsetsf: Modeling long-term time series forecasting with 1k parameters. *arXiv preprint arXiv:2405.00946*, 2024.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and JUN ZHOU. Timemixer: Decomposable multiscale mixing for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024b.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023b.

Donghao Luo and Xue Wang. Deformabletst: Transformer for time series forecasting without over-reliance on patching. *Advances in Neural Information Processing Systems*, 37:88003–88044, 2024.

Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-tao Xia. Timebridge: Non-stationarity matters for long-term time series forecasting. *arXiv preprint arXiv:2410.04442*, 2024.

A Extended Related Work

In this appendix, we provide a more detailed discussion of the work most related to BucpTSF. We organize the literature into three lines: temporal representation reorganization, structured and frequency-aware token interaction, and horizon-aware forecasting with structured extrapolation.

A.1 Temporal Representation Reorganization for Time-Series Forecasting

Existing studies have improved forecasting performance by reorganizing input time-series representations from multiple perspectives, including decomposition, rearrangement, patching, and tokenization. Autoformer embeds series decomposition into the network and progressively separates trend and seasonal components, alleviating pattern entanglement in long-term forecasting Wu et al. [2021]. TimesNet reshapes one-dimensional time series into two-dimensional tensors and models both intra-period and inter-period variations, showing that complex temporal dynamics can become easier to model after appropriate structural reorganization Wu et al. [2022]. These methods suggest that how the raw sequence is reorganized already has a substantial influence on subsequent forecasting effectiveness.

Another line of work focuses on how tokens should be defined and organized. PatchTST partitions time series into fixed-length patches and treats each patch as a token, demonstrating that patch-level tokenization can preserve local semantics while reducing attention cost Nie et al. [2022]. iTransformer further changes the tokenization direction by treating variables rather than temporal positions as tokens, showing that token granularity and organization are themselves critical design choices for forecasting Liu et al. [2023a]. For exogenous-variable settings, TimeXer combines patch-wise self-attention and variate-wise cross-attention, and introduces a global endogenous token to bridge exogenous information, further highlighting the importance of token-space organization for efficient information integration Wang et al. [2024a].

More recent work has begun to treat tokenization itself as a core modeling object. WaveToken constructs a wavelet-based tokenizer, allowing representations to be learned in a time-localized frequency space rather than through uniform partitioning in the original time domain Masserano et al. [2025]. Overall, prior methods improve temporal representation quality from different angles, but their primary goal is still to learn better representations. In contrast, BucpTSF reorganizes the raw sequence into a hierarchically heterogeneous token space so that subsequent modeling capacity can be allocated in a non-uniform and structure-aware manner.

A.2 Structured and Frequency-Aware Token Interaction

In terms of token interaction, existing forecasting models have mainly evolved along three directions: sparse connectivity, multi-scale structural modeling, and frequency-domain interaction. Informer introduces ProbSparse attention to reduce the time and memory cost of long-sequence attention, reflecting an early recognition that not all token interactions are equally important Zhou et al. [2021]. Pyraformer adopts a pyramidal attention graph to model temporal dependencies under different receptive fields, replacing dense all-to-all interaction with more organized structural connections Liu et al. [2021]. These methods suggest that token dependencies need not be modeled through uniform dense interaction.

Another line of work incorporates temporal priors and frequency structures into token interaction. ETSformer integrates exponential smoothing into Transformer-style forecasting and explicitly models level, growth, and seasonality through exponential smoothing attention and frequency attention Woo et al. [2022]. FEDformer extends interaction into the frequency domain, captures global structures in spectral representations, and combines decomposition-based modeling of trend and residual components Zhou et al. [2022]. Fredformer further points out the frequency bias of Transformer-based forecasters, namely their tendency to prefer dominant low-frequency components while overlooking low-amplitude but informative high-frequency patterns Piao et al. [2024]. This observation suggests that token interaction should not only be efficient, but also explicitly frequency-selective.

More recent research explores more flexible forms of structured interaction. Pathformer dynamically selects modeling pathways across multi-scale patches while considering both temporal resolution and temporal distance, reflecting a shift from static global mixing to more structured, conditional, and path-aware interaction design Chen et al. [2024]. Overall, existing methods have significantly improved token interaction, but many still rely on relatively heavy attention mechanisms or complicated multi-branch architectures. Different from these works, BucpTSF uses a lightweight structured interaction operator with explicit spectral selectivity, aiming to more directly allocate modeling capacity to structurally meaningful cross-token relations.

A.3 Horizon-Aware Forecasting and Structured Extrapolation

For long-horizon forecasting, an increasing number of studies have begun to revisit a key question: should different forecasting horizons share the same prediction complexity? One representative line of work emphasizes the importance of structured extrapolation bias. DLinear shows that even a very simple direct multi-step forecasting form can achieve strong long-horizon performance when equipped with appropriate linear inductive bias through trend-residual decomposition Zeng et al. [2023]. Similarly, TSMixer replaces heavy attention modules with MLP-based mixing along temporal and feature dimensions, further suggesting that forecasting performance may depend less on increasingly complex backbones and more on how the prediction mapping itself is organized Ekambaram et al. [2023].

Another line of work supports a similar view from frequency-domain or low-resource perspectives. FITS formulates forecasting as interpolation in the complex frequency domain and obtains competitive performance with very few parameters by discarding less important high-frequency components, suggesting that long-range prediction often benefits from compact and structured extrapolation mechanisms Xu et al. [2023]. MixLinear combines intra-segment and inter-segment variations in the time domain with low-dimensional frequency filtering, further supporting the view that long-horizon forecasting benefits more from structured bias than from uniformly high complexity Ma et al. [2024]. OLinear learns linear forecasting in an orthogonal transform domain, reflecting a recent shift from asking whether linear extrapolation works to asking in which representation space linear extrapolation is most effective Yue et al. [2025].

Taken together, these studies advance long-term forecasting from the perspectives of linear modeling, frequency-domain extrapolation, and lightweight prediction architectures, and jointly suggest that different horizons may require different modeling strategies. However, most of them still apply a relatively uniform prediction form across all future steps. In contrast, BucpTSF explicitly introduces horizon-conditioned structured extrapolation, allowing the model to retain stronger dynamic modeling capacity for near horizons while injecting stronger structural extrapolation bias for far horizons, thereby improving the stability and generalization of long-range forecasting.

B Extended Experiments

Computational resources. All experiments are conducted on NVIDIA RTX 4080 16GB and NVIDIA A800 80GB GPUs. We mainly use RTX 4080 for smaller datasets and ablation studies, while A800 is used for datasets with higher dimensionality or longer sequences to ensure stable training. All models are trained and evaluated under the same experimental protocol for fair comparison.

Broader impacts. BucpTSF aims to improve the accuracy, stability, and computational efficiency of time-series forecasting models under complex dynamic scenarios, which may bring positive impacts in real-world applications. For example, in energy management, traffic scheduling, environmental monitoring, and industrial operation, more reliable forecasting models can help improve resource utilization, reducea operational costs, and support timely risk warning. In addition, the perspective of information-density-driven non-uniform computation allocation may provide useful insights for designing more efficient time-series models.

At the same time, time-series forecasting models may also introduce potential risks when deployed in practice. First, forecasting results may be used in high-impact decision-making scenarios, such as power dispatching, traffic control, or financial analysis, where model errors, data bias, or distribution shifts could lead to unreliable decisions. Second, if the training data contain sensitive behavioral patterns or personal information, privacy protection and data compliance should be carefully considered. Moreover, improving forecasting efficiency may reduce the cost of large-scale deployment, but it may also encourage over-reliance on automated prediction systems. Therefore, in practical applications, model outputs should be used carefully together with domain expertise, anomaly monitoring, and human oversight.

B.1 Dataset Descriptions

Table 6 summarizes the datasets used in our experiments. The benchmark suite covers three representative forecasting settings: standard long-term forecasting, low-dimensional forecasting with complex dependency patterns, and short-term traffic forecasting. For standard long-term forecasting, we include widely used datasets from the ETT family, Weather, Electricity, and Traffic, which cover different domains, sampling frequencies, variable dimensions, and temporal dynamics. These datasets are used to evaluate whether BucpTSF can consistently model both global trends and local fluctuations under conventional long-horizon forecasting protocols.

In addition to standard benchmarks, we further evaluate BucpTSF on several low-dimensional datasets, including AQShunyi, AQWan, CzeLan, Temp, and Wind. These datasets have smaller or moderate variable dimensions, but may still contain complex local dynamics, heterogeneous temporal patterns, or non-stationary variations. They are therefore useful for examining whether the proposed information-density-driven design remains effective when the challenge comes less from the number of variables and more from the structure of temporal dependencies.

For short-term forecasting, we use the PEMS series, including PEMS03, PEMS04, PEMS07, and PEMS08. These datasets are collected from traffic sensor networks and exhibit strong high-frequency dynamics, short-range temporal dependencies, and cross-variable coupling. We evaluate them under prediction lengths $\{12, 24, 48, 96\}$ to test the ability of BucpTSF to capture complex local patterns in multivariate short-term forecasting.

Overall, the selected datasets span different domains, dimensions, sampling frequencies, and forecasting horizons. This diversity allows us to evaluate BucpTSF under a broad range of temporal structures and to verify whether its non-uniform computation allocation mechanism generalizes beyond a single dataset type or prediction setting. All datasets used in this work are publicly available benchmark datasets. We follow their commonly adopted academic usage and preprocessing protocols, and cite the corresponding original papers or benchmark sources. No proprietary or restricted-access datasets are used.

B.2 Hyperparameter Settings

To facilitate reproducibility, we report the dataset-specific hyperparameter settings of BucpTSF in Table 7. These hyperparameters correspond to the core implementation choices shared by the

Table 6: Dataset descriptions used in our experiments. *Dim* denotes the number of variables, and *Horizon* denotes the prediction lengths. *Size* gives the reported (train, validation, test) split where available and the total sequence length otherwise. For PEMS, the split counts are raw chronological time steps before sliding-window construction, following the 6:2:2 protocol of iTransformer.

Task	Dataset	Dim	Horizon	Size	Frequency	Information
Long-term Forecasting	ETTm1	7	$\{96, 192, 336, 720\}$	$(34465, 11521, 11521)$	15min	Temperature
	ETTm2	7	$\{96, 192, 336, 720\}$	$(34465, 11521, 11521)$	15min	Temperature
	ETTh1	7	$\{96, 192, 336, 720\}$	$(8545, 2881, 2881)$	1hour	Temperature
	ETTh2	7	$\{96, 192, 336, 720\}$	$(8545, 2881, 2881)$	1hour	Temperature
	Electricity	321	$\{96, 192, 336, 720\}$	$(18317, 2633, 5261)$	1hour	Electricity
	Traffic	862	$\{96, 192, 336, 720\}$	$(12185, 1757, 3509)$	1hour	Transportation
	Weather	21	$\{96, 192, 336, 720\}$	$(36792, 5271, 10540)$	10min	Weather
Low-dimensional Forecasting	AQShunyi	11	$\{96, 192, 336, 720\}$	35064	1hour	Air quality
	AQWan	11	$\{96, 192, 336, 720\}$	35064	1hour	Air quality
	CzeLan	11	$\{96, 192, 336, 720\}$	19934	30min	Low-dimensional series
	Temp	184	$\{96, 192, 336, 720\}$	11688	3hours	Temperature
	Wind	7	$\{96, 192, 336, 720\}$	48673	15min	Wind
Short-term Forecasting	PEMS03	358	$\{12, 24, 48, 96\}$	$(15724, 5242, 5242)$	5min	Transportation
	PEMS04	307	$\{12, 24, 48, 96\}$	$(10195, 3398, 3399)$	5min	Transportation
	PEMS07	883	$\{12, 24, 48, 96\}$	$(16934, 5645, 5645)$	5min	Transportation
	PEMS08	170	$\{12, 24, 48, 96\}$	$(10713, 3571, 3572)$	5min	Transportation

Table 7: Dataset-specific hyperparameter settings of BucpTSF.

Dataset	d_model	tfactor	dfactor	wavelet	level	patch_len	stride	batch_size	lr	lradj	dropout	emb_drop	patience	epochs
ETTh1	256	5	8	db2	2	8	8	256	2.000e-4	type3	0.40	0.10	12	60
ETTh2	256	5	5	db2	2	16	8	256	4.000e-4	type3	0.00	0.10	12	30
ETTm1	256	5	3	db2	1	48	24	256	1.000e-3	type3	0.40	0.20	12	80
ETTm2	256	3	8	bior3.1	1	48	24	256	7.000e-4	type3	0.40	0.00	12	80
Weather	256	3	7	db3	2	16	8	32	9.000e-4	type3	0.40	0.10	12	60
Electricity	32	3	5	sym3	2	16	8	32	3.000e-3	type3	0.10	0.00	12	100
Traffic	16	3	5	db3	1	16	8	16	1.000e-3	type3	0.05	0.05	12	60
AQShunyi	256	3	5	db2	1	48	32	256	9.000e-4	type3	0.40	0.00	3	60
AQWan	256	3	5	db2	1	48	32	256	9.000e-4	type3	0.40	0.00	3	60
CzeLan	8	3	7	db2	2	48	8	32	3.000e-3	type3	0.10	0.00	12	100
Temp	8	5	8	db2	2	8	8	256	2.000e-4	type3	0.40	0.10	12	60
Wind	256	5	5	db3	2	16	8	256	2.000e-4	type3	0.05	0.20	12	30
PEMS03	32	3	5	sym3	2	16	8	32	3.000e-3	type3	0.10	0.00	12	100
PEMS04	32	3	5	sym3	2	16	8	32	3.000e-3	type3	0.10	0.00	12	100
PEMS07	32	3	5	sym3	2	16	8	32	3.000e-3	type3	0.10	0.00	12	100
PEMS08	64	3	5	sym3	2	16	8	32	3.000e-3	type3	0.10	0.00	12	100

methodological design and the code, including the token embedding dimension, multiresolution decomposition configuration, patching strategy, optimization settings, and regularization parameters.

B.3 Extended Long-Term Forecasting Results

In the main text, we report the averaged long-term forecasting performance across all prediction horizons to provide a compact comparison with existing baselines. To complement the averaged results, Table 8 provides the complete horizon-wise performance of BucpTSF on the traditional long-term forecasting benchmarks, including ETTm1, ETTm2, ETTh1, ETTh2, Weather, Traffic, and Electricity.

Overall, BucpTSF maintains stable performance across different prediction lengths. As expected, the prediction error generally increases as the forecasting horizon becomes longer, since long-horizon forecasting requires the model to extrapolate more uncertain future dynamics. Nevertheless, BucpTSF shows a relatively smooth degradation from short to long horizons on most datasets, indicating that the proposed horizon-conditioned structured extrapolation mechanism helps improve long-range forecasting stability.

Across the ETT datasets, BucpTSF obtains consistently strong results under all horizons, demonstrating its ability to capture both short-term fluctuations and long-term temporal structures. On Weather, Traffic, and Electricity, the model also remains stable as the horizon increases, suggesting that information-density-driven representation reorganization and frequency-selective token interaction are effective across different variable scales and temporal patterns. These detailed results further support the averaged comparisons reported in the main text and show that the advantage of BucpTSF is not limited to a specific prediction length.

Table 8: The long-term prediction performance of BucpTSF on traditional datasets.

		Ours		2025(l_{weight})		2024(l_{weight})		2025		2023–2024							
Models	H	BucpTSF(ours)		TimeBase		SparseTSF		AMD		iTransformer		DeformableTST		TimeMixer		PatchTST	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.277**	**0.333**	0.311	0.351	0.314	0.359	0.284	0.339	0.300	0.353	0.291	0.347	0.293	0.345	0.293	0.346
	192	**0.321**	**0.361**	0.338	0.371	0.348	0.376	0.322	0.362	0.345	0.382	0.325	0.372	0.335	0.372	0.333	0.370
	336	**0.356**	0.382	0.364	0.386	0.368	0.386	0.361	0.383	0.374	0.398	0.359	0.390	0.368	0.386	0.369	**0.369**
	720	0.416	0.417	**0.413**	0.414	0.419	**0.413**	0.421	0.418	0.429	0.430	0.418	0.423	0.426	0.417	0.416	0.420
	Avg	**0.343**	**0.373**	0.357	0.381	0.362	0.384	0.347	0.376	0.362	0.391	0.348	0.383	0.356	0.380	0.353	0.376
ETTm2	96	**0.160**	**0.249**	0.162	0.256	0.167	0.259	0.167	0.258	0.175	0.266	0.169	0.258	0.165	0.256	0.166	0.256
	192	**0.215**	**0.289**	0.218	0.293	0.219	0.297	0.221	0.295	0.242	0.312	0.229	0.299	0.225	0.298	0.223	0.296
	336	0.271	**0.325**	**0.270**	0.328	0.271	0.330	**0.270**	0.327	0.282	0.340	0.280	0.333	0.277	0.332	0.274	0.329
	720	0.352	**0.379**	0.352	0.380	0.353	0.380	0.356	0.382	0.378	0.398	**0.349**	0.384	0.360	0.387	0.362	0.385
	Avg	**0.250**	**0.311**	0.251	0.314	0.253	0.317	0.254	0.316	0.269	0.329	0.257	0.319	0.257	0.318	0.256	0.317
ETTh1	96	**0.347**	**0.384**	0.349	**0.384**	0.362	0.389	0.369	0.397	0.386	0.405	0.369	0.396	0.372	0.401	0.370	0.400
	192	**0.380**	**0.406**	0.387	0.410	0.404	0.412	0.401	0.417	0.424	0.440	0.410	0.417	0.413	0.430	0.413	0.429
	336	0.399	0.432	0.408	0.418	0.435	0.428	0.418	0.427	0.449	0.460	**0.391**	**0.414**	0.438	0.450	0.422	0.440
	720	**0.409**	**0.434**	0.439	0.446	0.426	0.448	0.439	0.454	0.495	0.487	0.447	0.464	0.486	0.484	0.447	0.468
	Avg	**0.384**	**0.414**	0.396	0.415	0.407	0.419	0.407	0.424	0.439	0.448	0.404	0.423	0.427	0.441	0.413	0.434
ETTh2	96	**0.263**	**0.331**	0.292	0.345	0.294	0.346	0.274	0.337	0.297	0.348	0.272	0.334	0.281	0.351	0.274	0.337
	192	**0.307**	**0.363**	0.339	0.387	0.340	0.377	0.351	0.383	0.371	0.403	0.325	0.369	0.349	0.387	0.314	0.382
	336	0.346	0.398	0.358	0.410	0.360	0.398	0.375	0.411	0.404	0.428	**0.319**	**0.373**	0.366	0.413	0.329	0.384
	720	0.396	0.432	0.400	0.448	0.383	0.425	0.402	0.438	0.424	0.444	0.395	0.433	0.401	0.436	**0.379**	**0.422**
	Avg	0.328	0.381	0.347	0.398	0.344	0.387	0.351	0.392	0.374	0.406	0.328	**0.377**	0.349	0.397	**0.324**	0.381
Weather	96	**0.144**	**0.188**	0.146	0.198	0.174	0.231	0.145	0.198	0.159	0.208	0.146	0.198	0.147	0.198	0.149	0.198
	192	**0.185**	**0.230**	**0.185**	0.241	0.216	0.267	0.190	0.240	0.200	0.248	0.191	0.239	0.192	0.243	0.194	0.241
	336	0.238	**0.272**	**0.236**	0.281	0.260	0.299	0.242	0.280	0.253	0.289	0.241	0.280	0.247	0.284	0.245	0.282
	720	0.310	**0.326**	**0.309**	0.331	0.325	0.345	0.315	0.332	0.321	0.338	0.310	0.331	0.318	0.330	0.314	0.334
	Avg	**0.219**	**0.254**	**0.219**	0.263	0.244	0.286	0.223	0.263	0.233	0.271	0.222	0.262	0.226	0.264	0.226	0.264
Traffic	96	**0.355**	**0.248**	0.394	0.267	0.389	0.268	0.366	0.259	0.363	0.265	**0.355**	0.261	0.369	0.257	0.360	0.249
	192	**0.372**	**0.255**	0.403	0.271	0.399	0.272	0.381	0.265	0.385	0.273	0.380	0.271	0.400	0.272	0.379	0.256
	336	**0.388**	0.268	0.417	0.278	0.417	0.279	0.397	0.273	0.396	0.277	0.393	0.281	0.407	0.272	0.392	**0.264**
	720	**0.426**	0.288	0.456	0.298	0.449	0.299	0.430	0.292	0.445	0.312	0.434	0.300	0.461	0.316	0.432	**0.286**
	Avg	**0.385**	**0.265**	0.418	0.279	0.414	0.280	0.394	0.272	0.397	0.282	0.391	0.278	0.409	0.279	0.391	**0.264**
Electricity	96	**0.128**	**0.222**	0.139	0.231	0.139	0.239	0.129	0.225	0.138	0.237	0.132	0.234	0.153	0.256	0.129	**0.222**
	192	**0.144**	**0.236**	0.153	0.245	0.155	0.250	0.148	0.242	0.157	0.256	0.148	0.248	0.168	0.269	0.147	0.240
	336	**0.160**	**0.255**	0.169	0.262	0.171	0.260	0.164	0.258	0.167	0.264	0.165	0.266	0.189	0.291	0.163	0.259
	720	0.196	0.288	0.207	0.294	0.208	0.300	0.195	0.289	**0.194**	**0.286**	0.197	0.296	0.228	0.320	0.197	0.290
	Avg	**0.157**	**0.250**	0.167	0.258	0.168	0.262	0.159	0.254	0.164	0.261	0.161	0.261	0.185	0.284	0.159	0.253

Table 9: Long-term forecasting results on additional datasets. The best results are highlighted in bold.

Models	Metric	BucpTSF(ours)		TimeBase (2025)		SparseTSF (2024)		TimeMixer (2024)		FITS (2024)		iTransformer (2024)		DLinear (2023)		PatchTST (2023)		TimesNet (2023)	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
AQshunyi	96	**0.614**	**0.463**	0.629	0.489	0.751	0.553	0.703	0.534	0.759	0.555	0.686	0.508	0.751	0.579	0.643	0.476	0.759	0.560
	192	**0.656**	**0.480**	0.661	0.500	0.774	0.551	0.720	0.524	0.774	0.553	0.728	0.520	0.771	0.571	0.692	0.501	0.785	0.557
	336	**0.681**	**0.497**	0.683	0.510	0.752	0.544	0.747	0.546	0.755	0.544	0.732	0.529	0.758	0.580	0.695	0.514	0.802	0.566
	720	**0.725**	**0.518**	**0.725**	0.523	0.763	0.537	0.772	0.541	0.763	0.538	0.744	0.524	0.748	0.559	0.732	0.520	0.807	0.554
	Avg	**0.669**	**0.490**	0.675	0.506	0.760	0.546	0.736	0.536	0.763	0.548	0.723	0.520	0.757	0.572	0.691	0.503	0.788	0.559
AQWan	96	**0.708**	**0.454**	0.726	0.477	0.758	0.486	0.762	0.484	0.795	0.510	0.798	0.511	0.835	0.531	0.735	0.467	0.828	0.527
	192	**0.752**	**0.470**	0.761	0.492	0.851	0.542	0.779	0.492	0.765	0.485	0.785	0.501	0.884	0.558	0.792	0.500	0.807	0.510
	336	**0.780**	**0.487**	0.783	0.502	0.810	0.531	0.857	0.559	0.861	0.564	0.818	0.537	0.842	0.550	0.804	0.525	0.810	0.529
	720	0.845	**0.511**	0.847	0.524	0.883	0.546	0.891	0.549	**0.831**	0.517	0.971	0.604	0.972	0.600	0.909	0.561	0.982	0.606
	Avg	**0.771**	**0.481**	0.779	0.499	0.826	0.526	0.822	0.521	0.813	0.519	0.843	0.538	0.883	0.560	0.810	0.513	0.857	0.543
CzeLan	96	**0.168**	**0.213**	0.177	0.224	0.204	0.263	0.182	0.233	0.208	0.268	0.202	0.259	0.187	0.239	0.184	0.236	0.204	0.262
	192	**0.185**	**0.230**	0.211	0.250	0.227	0.274	0.216	0.259	0.242	0.292	0.233	0.282	0.226	0.271	0.214	0.257	0.256	0.308
	336	**0.210**	**0.254**	0.243	0.276	0.239	0.305	0.246	0.312	0.247	0.316	0.239	0.305	0.238	0.301	0.229	0.290	0.274	0.347
	720	**0.265**	**0.290**	0.269	0.297	0.292	0.325	0.291	0.322	0.303	0.338	0.305	0.340	0.315	0.349	0.288	0.319	0.299	0.332
	Avg	**0.207**	**0.247**	0.225	0.262	0.241	0.292	0.234	0.282	0.250	0.304	0.245	0.297	0.242	0.290	0.229	0.276	0.258	0.312
Temp	96	**0.134**	**0.281**	0.142	0.286	0.149	0.300	0.158	0.316	0.168	0.338	0.155	0.310	0.165	0.329	0.145	0.291	0.155	0.309
	192	**0.147**	**0.296**	0.155	0.305	0.168	0.323	0.163	0.312	0.181	0.351	0.168	0.324	0.174	0.335	0.166	0.319	0.187	0.358
	336	**0.169**	**0.318**	0.176	0.323	0.220	0.373	0.249	0.421	0.223	0.381	0.232	0.395	0.246	0.417	0.221	0.374	0.236	0.400
	720	**0.273**	**0.435**	0.274	0.437	0.642	0.654	0.637	0.643	0.689	0.699	0.652	0.663	0.584	0.590	0.632	0.639	0.636	0.642
	Avg	**0.181**	**0.333**	0.187	0.338	0.295	0.413	0.302	0.423	0.315	0.442	0.302	0.423	0.292	0.418	0.291	0.406	0.304	0.427
Wind	96	**0.775**	0.608	0.785	0.626	0.796	0.589	0.908	0.669	0.787	**0.581**	0.824	0.609	0.824	0.607	0.802	0.591	0.892	0.657
	192	**0.901**	0.682	0.902	0.690	0.949	0.698	1.040	0.764	0.910	**0.674**	1.054	0.779	0.961	0.707	0.944	0.694	1.008	0.741
	336	**0.983**	**0.725**	0.997	0.737	1.182	0.851	1.039	0.747	1.100	0.795	1.025	0.742	1.022	0.734	1.043	0.750	1.185	0.853
	720	**1.073**	**0.766**	1.075	0.782	1.127	0.808	1.189	0.851	1.295	0.932	1.196	0.863	1.266	0.907	1.084	0.776	1.320	0.946
	Avg	**0.933**	**0.695**	0.940	0.709	1.014	0.737	1.044	0.758	1.023	0.746	1.025	0.748	1.018	0.739	0.968	0.703	1.101	0.799

B.4 Short-Term Forecasting Results on PEMS

Table 10 reports the full short-term forecasting results on the PEMS datasets under prediction lengths $\{12, 24, 48, 96\}$. Overall, BucpTSF achieves the best average performance on all four datasets in terms of both MSE and MAE, demonstrating its effectiveness in short-term traffic forecasting scenarios. Compared with Transformer-based and MLP-based baselines, BucpTSF consistently obtains lower errors across most forecasting horizons, indicating that the proposed non-uniform computation allocation is not only effective for long-term forecasting, but also beneficial for modeling short-range temporal dependencies.

On PEMS03 and PEMS04, BucpTSF achieves the best results at nearly all horizons. The improvements are especially clear in the average results, where BucpTSF obtains lower MSE and MAE than Twins, iTransformer, PatchTST, Crossformer, TiDE, and TimesNet. This suggests that BucpTSF can more effectively capture local traffic dynamics and short-term temporal variations. Since short-term traffic forecasting is highly sensitive to abrupt changes and local fluctuations, the strong performance of BucpTSF indicates that representation reorganization and frequency-selective interaction help preserve informative local structures while avoiding redundant uniform computation.

On PEMS07 and PEMS08, BucpTSF also maintains highly competitive and generally superior performance. For PEMS07, BucpTSF achieves the best average MSE and MAE, and consistently performs best across most horizons. For PEMS08, BucpTSF obtains the best average results, although Twins slightly outperforms BucpTSF on the MSE metric at horizon 96. This indicates that BucpTSF remains robust as the forecasting horizon increases, while also showing that some baselines may still be competitive on specific long-horizon short-term settings. Nevertheless, the average results confirm that BucpTSF provides a more stable overall performance across different prediction lengths.

These results further support the generality of BucpTSF. Unlike methods that rely on uniform token mixing or fixed prediction structures, BucpTSF adaptively allocates modeling capacity across temporal representations, frequency components, and forecasting horizons. Such a design is particularly useful for traffic forecasting, where different sensors and time steps may exhibit heterogeneous temporal patterns. Therefore, the short-term forecasting results in Table 10 complement the long-term forecasting experiments and verify that BucpTSF can achieve strong performance under both short-range and long-range prediction settings.

Table 10: Short-term forecasting performance on the PEMS datasets with prediction lengths of $\{12, 24, 48, 96\}$. The best results are highlighted in bold.

Models	**H**	BucpTSF(ours)		Twins*		iTrans*		RLinear		PatchTST		Cross*		TiDE		TimesNet	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS03	12	**0.060**	**0.163**	0.065	0.169	0.069	0.175	0.126	0.236	0.099	0.216	0.090	0.203	0.178	0.305	0.085	0.192
	24	**0.075**	**0.180**	0.086	0.196	0.097	0.208	0.246	0.334	0.142	0.312	0.121	0.202	0.371	0.318	0.118	0.223
	48	**0.098**	**0.208**	0.121	0.234	0.131	0.243	0.551	0.529	0.211	0.319	0.202	0.317	0.463	0.355	0.206	0.315
	96	**0.127**	**0.227**	0.165	0.276	0.168	0.279	1.057	0.787	0.269	0.370	0.262	0.367	0.490	0.539	0.228	0.317
	Avg	**0.090**	**0.195**	0.109	0.219	0.116	0.226	0.495	0.472	0.180	0.304	0.169	0.272	0.376	0.379	0.159	0.262
PEMS04	12	**0.072**	**0.175**	0.077	0.181	0.081	0.188	0.138	0.252	0.105	0.224	0.098	0.218	0.219	0.340	0.087	0.195
	24	**0.086**	**0.190**	0.095	0.204	0.099	0.211	0.258	0.348	0.153	0.275	0.131	0.256	0.292	0.330	0.108	0.215
	48	**0.103**	**0.208**	0.120	0.231	0.133	0.247	0.572	0.544	0.229	0.339	0.205	0.336	0.409	0.478	0.136	0.250
	96	**0.128**	**0.239**	0.150	0.261	0.172	0.283	1.137	0.820	0.291	0.389	0.402	0.457	0.492	0.532	0.190	0.303
	Avg	**0.097**	**0.203**	0.111	0.219	0.121	0.232	0.526	0.491	0.195	0.307	0.209	0.317	0.353	0.420	0.130	0.241
PEMS07	12	**0.058**	**0.155**	0.060	0.158	0.067	0.167	0.118	0.235	0.095	0.207	0.094	0.200	0.173	0.304	0.082	0.181
	24	**0.073**	**0.172**	0.079	0.181	0.086	0.189	0.242	0.341	0.150	0.262	0.139	0.247	0.271	0.383	0.101	0.204
	48	**0.101**	**0.204**	0.104	0.209	0.110	0.214	0.562	0.541	0.253	0.340	0.311	0.369	0.446	0.495	0.134	0.238
	96	**0.125**	**0.236**	0.132	**0.236**	0.138	0.244	1.096	0.795	0.346	0.404	0.396	0.442	0.628	0.577	0.181	0.279
	Avg	**0.089**	**0.192**	0.094	0.196	0.100	0.204	0.505	0.478	0.211	0.303	0.235	0.315	0.380	0.440	0.125	0.226
PEMS08	12	**0.073**	**0.168**	0.075	0.174	0.080	0.183	0.133	0.247	0.168	0.232	0.165	0.214	0.227	0.343	0.112	0.212
	24	**0.096**	**0.187**	0.106	0.206	0.118	0.221	0.249	0.343	0.224	0.281	0.215	0.260	0.318	0.409	0.141	0.238
	48	**0.132**	**0.202**	0.167	0.258	0.186	0.265	0.569	0.544	0.321	0.354	0.315	0.355	0.497	0.510	0.198	0.283
	96	0.188	**0.235**	**0.184**	0.251	0.221	0.267	1.166	0.814	0.408	0.417	0.377	0.397	0.721	0.592	0.320	0.351
	Avg	**0.122**	**0.198**	0.133	0.222	0.151	0.234	0.529	0.487	0.280	0.321	0.268	0.307	0.441	0.464	0.193	0.271

* indicates a Transformer-based model.

B.5 Ablation

Detailed ablation protocol. To more systematically validate the roles of the components in BucpTSF, we further report full horizon-wise ablation results in the appendix (Table 11). Unlike the main text, which only reports the average MSE across forecasting horizons, the appendix presents the detailed results of each dataset at all forecasting lengths, enabling a finer-grained analysis of how different modules contribute under short-, medium-, and long-horizon settings. This organization, where the main text provides averaged results while the appendix provides the full horizon-wise results, allows us to present the overall conclusion compactly in the main paper while avoiding the risk that averaged values obscure the variation of model behavior across different horizons.

The ablation design in this section is motivated by two main considerations. First, we aim to evaluate the individual contributions of RR, FSI, and HSE, as well as their combinational effects. To this end, we adopt a compositional-style ablation protocol, starting from *Base* and progressively introducing

Table 11: Full ablation results of BucpTSF across all forecasting horizons in terms of MSE (lower is better). For ETTh1, ETTm1, and AQWan, we report results on horizons $\{96, 192, 336, 720\}$; for PEMS03, we report results on horizons $\{12, 24, 48, 96\}$. Base denotes the minimal BucpTSF backbone with single-scale patch embedding, uniform token mixing, and a shared prediction head. RR, FSI, and HSE denote representation reorganization, frequency-selective interaction, and horizon-conditioned structured extrapolation, respectively.

Model	ETTh1				ETTm1				AQWan				PEMS03			
	96	192	336	720	96	192	336	720	96	192	336	720	12	24	48	96
Compositional ablation																
Base	0.407	0.445	0.471	0.514	0.339	0.374	0.428	0.471	0.808	0.842	0.860	0.903	0.112	0.149	0.187	0.223
+ RR	0.381	0.419	0.430	0.441	0.300	0.350	0.384	0.452	0.746	0.788	0.801	0.881	0.089	0.110	0.148	0.193
+ FSI	0.387	0.411	0.424	0.436	0.296	0.346	0.381	0.445	0.739	0.781	0.796	0.876	0.086	0.104	0.143	0.189
+ HSE	0.391	0.425	0.437	0.445	0.309	0.359	0.396	0.457	0.758	0.792	0.812	0.869	0.094	0.112	0.152	0.199
+ RR + FSI	0.362	0.391	0.419	0.423	0.286	0.332	0.369	0.431	0.723	0.769	0.787	0.857	0.074	0.092	0.130	0.177
+ RR + HSE	0.374	0.401	0.420	0.429	0.293	0.340	0.376	0.436	0.734	0.774	0.797	0.870	0.081	0.101	0.136	0.184
+ FSI + HSE	0.369	0.394	0.412	0.421	0.285	0.334	0.371	0.433	0.725	0.773	0.793	0.867	0.077	0.096	0.135	0.179
Uniformity check																
Uniform-RR	0.361	0.389	0.410	0.416	0.282	0.329	0.365	0.427	0.719	0.766	0.792	0.859	0.073	0.093	0.125	0.171
Uniform-FSI	0.364	0.391	0.413	0.422	0.288	0.327	0.370	0.431	0.720	0.761	0.794	0.852	0.075	0.099	0.129	0.174
Uniform-HSE	0.356	0.383	0.404	0.417	0.280	0.329	0.362	0.422	0.713	0.763	0.781	0.855	0.068	0.089	0.123	0.166
BucpTSF	**0.347**	**0.380**	**0.399**	**0.409**	**0.277**	**0.321**	**0.356**	**0.416**	**0.708**	**0.752**	**0.780**	**0.845**	**0.060**	**0.075**	**0.098**	**0.127**

RR, FSI, and HSE, followed by comparisons between pairwise combinations and the full model. Compared with a simple "w/o" strategy, this design is more suitable for analyzing whether the modules exhibit complementarity: if a module produces more substantial gains only when combined with other modules, then its role is not merely an isolated improvement, but rather a synergistic contribution with the other components. In other words, this protocol answers not only whether each individual module is effective, but also whether the full model is superior to any of its partial combinations, thereby providing stronger support for the overall design motivation of BucpTSF.

Second, we further introduce three uniformized variants, namely Uniform-RR, Uniform-FSI, and Uniform-HSE, to examine whether the performance gains truly come from information-density-driven non-uniform computation allocation, rather than simply from adding extra structures. Specifically, Uniform-RR preserves the multi-level representation structure but removes hierarchical heterogeneity; Uniform-FSI preserves the frequency-domain interaction framework but removes frequency selectivity; and Uniform-HSE preserves the dual-branch prediction structure but removes the horizon-varying adaptive fusion. By comparing these uniformized variants with the full model, we can more directly verify the central claim of this work: the advantage of BucpTSF does not merely come from "using some form of representation reorganization, frequency-domain interaction, or dual-branch prediction," but from the non-uniform modeling capability jointly realized by these modules across temporal positions, frequency structures, and forecasting horizons.

In addition, there is another important reason for reporting the full horizon-wise results in the appendix: the effect of each module may vary across forecasting horizons. For example, the gain brought by HSE is often more pronounced at longer horizons, while FSI may play a stronger role on datasets with more complex high-frequency dynamics. Similarly, RR may become more critical on datasets with higher dimensionality or stronger structural heterogeneity. Therefore, although averaged results are sufficient to support the main conclusions in the paper, they are not sufficient to fully reveal the detailed impact of different modules under different settings. The complete appendix tables provide this fine-grained evidence and allow a more comprehensive analysis of the collaborative effects of BucpTSF at the representation, interaction, and prediction levels.

Overall, the full ablation results in the appendix are intended to further support the proposed design from three aspects: **(i)** single-module and two-module combinations validate the independent contributions and complementarity of RR, FSI, and HSE; **(ii)** the uniform variants verify whether information-density-driven non-uniform computation allocation is the key source of the performance gain; and **(iii)** the full horizon-wise results reveal how the role of each module varies across forecasting horizons and dataset characteristics. Based on this protocol, Table 11 presents the complete ablation results of BucpTSF under all experimental settings.

Table 12: Ablation on horizon gating functions in HSE. Results are reported at prediction horizon 96. Lower is better.

Gate Type	ETTh1	ETTm1	AQWan	PEMS03
Uniform Gate	0.353	0.283	0.714	0.135
Linear Gate	0.350	0.280	0.717	0.133
MLP Gate	0.355	0.281	0.710	0.131
Exp Gate (Ours)	**0.347**	**0.277**	**0.708**	**0.127**

Ablation on horizon gating functions. To justify the design of the horizon-conditioned structured extrapolation module, we further compare different gating functions for balancing the nonlinear backbone prediction and the low-rank structured extrapolation branch. Specifically, we consider four variants: *Uniform Gate*, which uses a shared constant fusion weight for all horizons; *Linear Gate*, which increases the extrapolation weight linearly with the forecasting horizon; *MLP Gate*, which predicts horizon-wise fusion weights from horizon embeddings; and the proposed *Exp Gate*, defined as $\beta(h) = 1 - \exp(-\lambda h)$.

As shown in Table 12, the proposed exponential gate achieves the best overall performance while introducing only one learnable scalar parameter. Compared with the uniform gate, the exponential gate explicitly models the increasing need for structured extrapolation at farther horizons. Compared with the linear gate, it provides a learnable and smoother horizon-dependent transition. Although the MLP gate is more expressive, it introduces additional parameters and can overfit or produce less stable horizon-wise fusion, especially on smaller datasets. These results suggest that the proposed exponential gate offers a favorable balance between simplicity, inductive bias, and forecasting accuracy.

B.6 Controlled information-density experiment.

To further validate the design motivation of our method from a mechanistic perspective, we construct a controlled synthetic forecasting experiment and systematically vary the information density of the input sequences. Specifically, we design four sequence settings, corresponding to low information density (*Low-ID*), stronger local dynamics (*Local-Dynamics*), more complex frequency structures (*Spectral-Complexity*), and high information density with both localized bursts and high-frequency complexity (*High-ID*). Based on these settings, we compare the relative gain of density-aware forecasting over uniform forecasting under different forecasting horizons. The results are shown in Fig. 5.

We observe that density-aware forecasting consistently outperforms uniform forecasting across all settings. More importantly, the gain is not constant, but increases systematically as the information density of the sequence becomes higher. Specifically, from *Low-ID* to *High-ID*, the relative improvement increases steadily across almost all horizons, indicating that information-density-driven non-uniform modeling becomes more beneficial when local temporal dynamics and frequency structures are more complex. Meanwhile, the relative gain generally decreases as the forecasting horizon becomes longer, but remains positive even at longer horizons. This suggests that the density-aware mechanism is not only effective for short-term forecasting, but can also provide sustained benefits over longer prediction ranges.

These results provide direct evidence for the core hypothesis of this work: when structural complexity becomes more non-uniform along temporal and spectral dimensions, uniform computation allocation becomes less efficient, whereas information-density-driven non-uniform modeling can better adapt to such heterogeneity and yield more stable performance improvements.

Table 13: Efficiency comparison between BucpTSF and TimeMixer on ETTh1. BucpTSF achieves lower forecasting errors with substantially fewer GFLOPs.

	BucpTSF			TimeMixer		
T	MSE	MAE	GFLOPs	MSE	MAE	GFLOPs
96	**0.347**	**0.384**	**0.239**	0.372	0.401	2.774
192	**0.380**	**0.406**	**0.257**	0.413	0.430	3.281
336	**0.399**	**0.432**	**0.343**	0.438	0.450	4.040
720	**0.409**	**0.434**	**0.505**	0.486	0.484	6.066

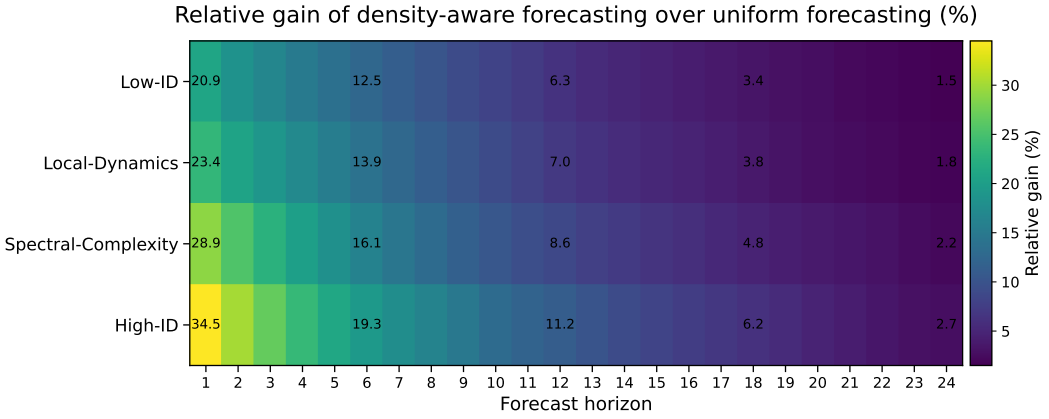

Figure 5: Relative gain of density-aware forecasting over uniform forecasting in the controlled synthetic experiment. The vertical axis corresponds to four synthetic sequence settings with increasing information density, while the horizontal axis corresponds to different forecasting horizons. As local dynamics and spectral complexity increase, density-aware forecasting brings increasingly larger gains, providing controlled experimental evidence for information-density-driven non-uniform computation allocation.

B.7 Computational Efficiency and Robustness

We further evaluate the computational efficiency of BucpTSF in terms of giga floating-point operations (GFLOPs), a hardware-independent metric for computational cost. Following the same setting as TimeMixer on the ETTh1 dataset, we report the MSE, MAE, and GFLOPs under prediction lengths of 96, 192, 336, and 720. The comparison is shown in Table 13.

As shown in Table 13, BucpTSF consistently requires substantially fewer GFLOPs than TimeMixer across all prediction horizons. Specifically, BucpTSF uses only 0.239, 0.257, 0.343, and 0.505 GFLOPs for horizons 96, 192, 336, and 720, respectively, while TimeMixer requires 2.774, 3.281, 4.040, and 6.066 GFLOPs. This corresponds to about a $12\times$ reduction in computational cost on average. Meanwhile, BucpTSF also achieves lower MSE and MAE across all horizons, indicating that the proposed information-density-driven non-uniform computation allocation improves computational efficiency without sacrificing forecasting accuracy.

B.8 Robustness to Random Seeds

To evaluate the robustness of BucpTSF, we run the model with three different random seeds and report the mean and standard deviation of MSE and MAE. Results are averaged over all forecasting horizons for each dataset: ETTh1, ETTh2, Traffic, ECL, and AQWan are averaged over $\{96, 192, 336, 720\}$, while PEMS03 is averaged over $\{12, 24, 48, 96\}$. As shown in Table 14, BucpTSF exhibits small performance variance across different random seeds, with all standard deviations no larger than 0.002. This indicates that the proposed framework is stable and robust to random initialization.

Table 14: Robustness of BucpTSF under three random seeds. Results are reported as mean±standard deviation over three runs. Lower is better.

Metric	ETTh1		ETTh2		Traffic	
	MSE	MAE	MSE	MAE	MSE	MAE
Mean±Std.	0.384±0.000	0.414±0.001	0.328±0.001	0.381±0.001	0.385±0.002	0.264±0.001
Metric	ECL		AQWan		PEMS03	
	MSE	MAE	MSE	MAE	MSE	MAE
Mean±Std.	0.157±0.002	0.250±0.002	0.771±0.002	0.481±0.001	0.090±0.000	0.195±0.000

Table 15: Occlusion-based predictive utility versus temporal/spectral proxy responses. Spearman rank correlations (higher is stronger alignment).

Dataset	Horizon	Temporal ρ	Spectral ρ
AQWan	96	0.397	0.433
AQWan	192	0.686	0.729
ETTh1	96	0.748	0.638
ETTh1	192	0.367	0.476
PEMS03	12	0.399	0.595
PEMS03	24	0.385	0.422
Mean	–	**0.497**	**0.549**

B.9 Alignment of Structural Proxies with Predictive Utility

To probe whether the model responses reflect forecasting usefulness rather than merely signal amplitude, we perform an occlusion-based diagnostic. For a temporal region or spectral mode k, define its utility as

$$U_k = \mathcal{L}(\hat{\mathbf{Y}}_{-k}, \mathbf{Y}) - \mathcal{L}(\hat{\mathbf{Y}}, \mathbf{Y}), \tag{19}$$

where $\hat{\mathbf{Y}}_{-k}$ is obtained by masking component k and $\mathcal{L}$ is the forecasting loss. We calculate Spearman's rank correlation between U_k and the corresponding temporal representation response or learned spectral-filter magnitude. Each setting uses 128 test samples with seed 42, following the rebuttal protocol.

All correlations are positive in the six examined settings, suggesting that more influential components generally elicit stronger learned responses. This supports the RR/FSI proxies as empirical mechanisms consistent with the proposed design principle, but does not imply identification of a unique information-density function or per-sample FLOP allocation.

B.10 Learned Horizon-Gate Behavior

The gate $\beta(h) = 1 - \exp(-\lambda_H h)$ uses a nonnegative λ_H trained separately for each dataset and forecast length H. Table 16 reports the learned transition rate, cumulative quantity $\lambda_H H$, final gate $\beta(H)$, and mean of $\beta(h)$ over $h = 1, \ldots, H$. While λ_H tends to decrease for larger H, the endpoint and mean gate can increase, reflecting a slower per-step transition over a longer forecasting interval. This monotonicity applies within each fitted gate, not to error improvements across different horizon lengths.

A monotone gate is an inductive bias, not a universal property of forecasting data: sequences with persistent bursts or abrupt late changes might benefit from non-monotone alternatives.

B.11 Matched Inference Efficiency

We additionally compare resource requirements under a matched ETTh1 setting (look-back length 512, prediction length 720, inference batch size 1, measured on the same GPU). Unlike the GFLOPs comparison in Table 13, this experiment reports parameter count, multiply–accumulate operations (MACs), peak inference memory, and end-to-end inference latency. These reflect fixed-graph efficiency rather than a sample-dependent computation mechanism.

Table 16: Learned horizon-gate behavior. Rounded values from the rebuttal; each H corresponds to a separately trained configuration.

Dataset	H	λ_H	$\lambda_H H$	$\beta(H)$	Mean β
ETTh1	96	0.00623	0.598	0.45	0.25
ETTh1	192	0.00452	0.868	0.58	0.33
ETTh1	336	0.00358	1.203	0.70	0.42
ETTh1	720	0.00238	1.714	0.82	0.52
Traffic	96	0.00832	0.799	0.55	0.31
Traffic	192	0.00562	1.079	0.66	0.39
Traffic	336	0.00425	1.428	0.76	0.47
Traffic	720	0.00295	2.124	0.88	0.59

Table 17: Matched ETTh1 inference-efficiency comparison ($L = 512$, $H = 720$, batch size 1). Measurements reported in the rebuttal.

Model	Params (M)	MACs (M)	Peak memory (MB)	Latency (s)
PatchTST	6.30	225	672	0.0100
BucpTSF	**3.91**	**182**	**79**	**0.0038**

Under this setting, BucpTSF has lower values for all four reported resource metrics than PatchTST. The comparison is specific to the chosen dataset, horizon, and evaluation environment and does not establish minimal compute across all forecasting methods or settings.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction summarize the main contributions and scope of BucpTSF, including information-density-driven non-uniform computation allocation, representation reorganization, frequency-selective interaction, and horizon-conditioned structured extrapolation. The claims are supported by the experimental results in the main paper and additional results in the appendix.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of BucpTSF in the paper, including its additional architectural complexity compared with minimal forecasting backbones and the fact that the empirical validation is conducted on commonly used forecasting benchmarks rather than all possible real-world deployment scenarios.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical results such as theorems or lemmas requiring proofs. The proposed method is validated primarily through empirical experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the model architecture, datasets, evaluation metrics, forecasting horizons, baseline comparisons, and implementation details needed to reproduce the main experimental results. Additional experimental details and full horizon-wise results are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide anonymized code and instructions in the supplemental material, including data preparation, training, and evaluation scripts for reproducing the main experimental results. The datasets used in this work are publicly available forecasting benchmarks.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the datasets, forecasting horizons, evaluation metrics, baseline methods, and training configurations used in the experiments. Additional implementation and hyperparameter details are included in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: We report the mean and standard deviation over multiple runs for the key experiments supporting the main claims. The corresponding results and calculation protocol are provided in the experimental section or appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the computational environment used for the experiments, including the GPU types and memory. We also report GFLOPs-based efficiency comparisons in the appendix to characterize the computational cost of BucpTSF.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses public forecasting datasets and does not involve human subjects, private personal data, or high-risk data collection. We have reviewed the NeurIPS Code of Ethics and believe the work conforms to it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses potential positive impacts of improved time-series forecasting, such as better decision-making in energy, traffic, and environmental monitoring, as well as possible risks from inaccurate predictions or misuse in high-stakes applications.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release high-risk generative models, scraped datasets, or assets with a high risk of misuse. Therefore, specific safeguards for controlled release are not applicable.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes].

Justification: The paper uses standard public time-series forecasting benchmark datasets and properly cites their original sources or commonly adopted benchmark protocols. To the best of our knowledge, these datasets are publicly available for academic research. We follow their standard usage and preprocessing protocols and do not redistribute any restricted or proprietary assets.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not introduce or release new datasets, benchmarks, or other standalone assets. The contribution is a forecasting model and its empirical evaluation on existing datasets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing, user studies, or experiments with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research, so IRB approval or equivalent review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development, experiments, and scientific claims in this work do not involve LLMs as an important, original, or non-standard component. Any use of language tools, if any, was limited to writing, editing, or formatting and did not affect the methodology or experimental results.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.