HARMOTC: HARMONIC TEMPORAL CONDITIONING WITH PHASE–AMPLITUDE SPECTRAL MODELING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing time-series forecasting methods primarily model pointwise dependencies in the time domain, making it difficult to preserve both periodic strength and temporal alignment. Moreover, periodic temporal cues are commonly treated only as auxiliary input features. To address these limitations, we propose HarMoTC, a framework that performs temporally conditioned phase–amplitude spectral modeling for multivariate time-series forecasting. HarMoTC explicitly decouples the amplitude and phase of the complex spectrum and employs learnable amplitude rescaling and phase calibration to preserve periodic strength while maintaining temporal alignment. Furthermore, we introduce a conditional spectral modulation mechanism driven by multi-harmonic Fourier encodings of temporal indices, enabling the model to directly modulate spectral transformations according to temporal context and thereby adapt its frequency responses. The resulting architecture jointly captures periodic structure, context-dependent spectral variations, and cross-variable dependencies. Experiments on multiple real-world datasets demonstrate that HarMoTC achieves competitive forecasting performance.

1 INTRODUCTION

Multivariate time-series forecasting plays an important role in a wide range of applications, including electricity management, traffic monitoring, weather forecasting, and industrial systems. Real-world time series typically exhibit multi-scale periodic patterns (Challu et al., 2023), cross-variable dependencies, and operating regimes that vary over time. These characteristics are particularly challenging in long-term forecasting, where inaccurate estimation of periodic strength or small temporal shifts may accumulate over the prediction horizon, resulting in misaligned peaks and troughs and distorted periodic patterns. Accurate forecasting therefore requires not only reducing point-wise prediction errors, but also preserving the structural properties of the future sequence.

Frequency-domain representations offer a compact view of periodic structures (Wang et al., 2025). The Fourier transform decomposes a sequence into frequency components; amplitude characterizes their strength, while phase determines their temporal alignment. Although amplitude and phase jointly determine the periodic shape of a sequence, they correspond to different sources of forecasting error. Amplitude errors indicate inaccurate periodic intensity, whereas phase errors lead to temporal misalignment of peaks and troughs. Explicitly separating and jointly modeling spectral amplitude and phase can therefore better preserve the structural characteristics of future sequences.

However, spectral patterns of real-world time series vary across temporal contexts. The same variable may exhibit different frequency responses at different hours, days, or operating stages. Exogenous inputs can improve time-series forecasting (Wang et al., 2024b). Calendar information should therefore be regarded not only as an additional input feature, but also as an informative prior for adapting spectral representations. Simply concatenating calendar features with time-domain observations does not allow them to directly control the spectral transformation, limiting the model's ability to capture context-dependent spectral variations.

Two complementary analyses on the ETTm1 training set characterize calendar-dependent spectral variation and the sensitivity of reconstruction to amplitude and phase perturbations, using the same look-back length $L = 96$ as in our forecasting experiments. Figure 1 (a) groups historical windows according to the hour-of-week state at their endpoints and aggregates normalized spectral amplitudes across variables. The pronounced variation over calendar states shows that the strength of the same

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

periodic components is context dependent. For Figure 1 (b), we automatically select the three globally highest-energy frequency components from the training split and perturb their amplitude scale α and phase shift δ. The reconstruction error forms a basin around the unperturbed point $(\alpha, \delta) = (1, 0)$ and increases when moving along either dashed axis. Thus, amplitude distortion and phase misalignment are distinct yet complementary sources of structural forecasting error.

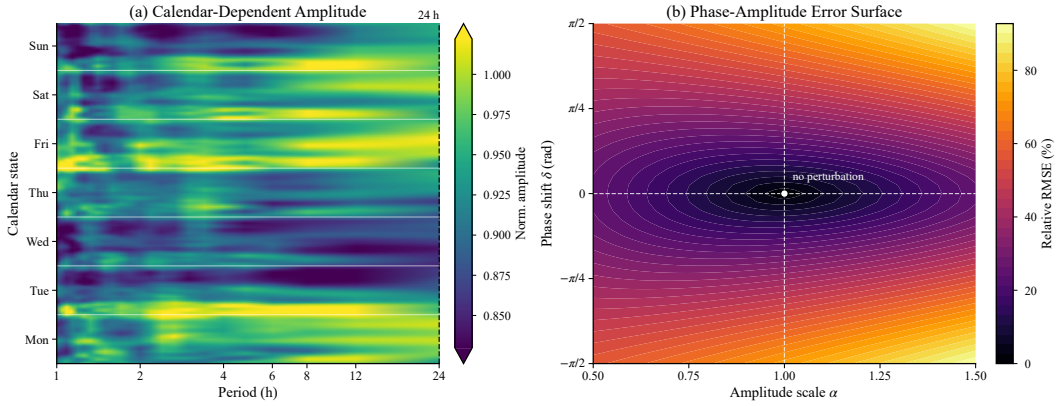

Figure 1: Training-set diagnostics on ETTm1 with $L = 96$. (a) Normalized spectral amplitude over 168 endpoint hour-of-week states; weekday labels denote the seven daily blocks. (b) Relative reconstruction RMSE after perturbing the amplitude scale α and phase shift δ of the three highest-energy components; dashed lines denote single-factor perturbations, and $(1, 0)$ is the unperturbed spectrum.

Motivated by these observations, we propose HarMoTC, a spectral reconstruction framework for multivariate time-series forecasting. HarMoTC is built on a simple frequency-domain forecasting backbone consisting of normalization, Fourier transformation, linear channel mixing, inverse transformation, and temporal projection. We augment this backbone with two core components. First, calendar-conditioned spectral modulation maps calendar indices such as hour and day into multi-harmonic Fourier representations and uses them to adapt the spectral transformation. Second, phase–amplitude decoupled spectral modeling explicitly decomposes the complex spectrum and reconstructs periodic structure through learnable amplitude rescaling and phase calibration. Together, the two components model variations in periodic strength, temporal alignment, and their dependence on temporal context, while the remaining operations serve as a shared forecasting backbone.

The main contributions of this work are summarized as follows:

- We introduce a conditional spectral modulation mechanism driven by Fourier calendar encodings, which incorporates calendar priors directly into frequency-domain representation learning and captures systematic spectral variations across temporal contexts.
- We propose phase–amplitude decoupled spectral modeling that explicitly separates and jointly reconstructs spectral amplitude and phase to model variations in periodic strength and temporal alignment.
- Extensive experiments on multiple real-world time-series datasets demonstrate the effectiveness and complementarity of the two proposed components, with HarMoTC achieving competitive forecasting performance.

2 Related Work

Long-term time-series forecasting. Long-term forecasting methods capture temporal and cross-variable patterns. Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), PatchTST (Nie et al., 2023), and iTransformer (Liu et al., 2024) employ sparse attention, decomposition, patches, and variate-wise attention, respectively. DLinear (Zeng et al., 2023) shows simple temporal mappings remain competitive. These methods provide strong time-domain baselines. However, these approaches

often represent changes in periodic strength and shifts in temporal alignment together, obscuring their separate effects on forecasts. For example, a recurring cycle may retain its frequency while changing in strength or shifting in time across calendar contexts. These variations motivate separate amplitude and phase controls in the spectral domain.

Frequency-domain forecasting. Frequency-domain approaches use spectral representations to capture global periodic structure. FEDformer (Zhou et al., 2022) combines frequency-enhanced modeling with decomposition, while TimesNet (Wu et al., 2023) organizes sequences by dominant periods. FreTS (Yi et al., 2023) and FITS (Xu et al., 2024) use spectral mappings and interpolation, respectively. These methods highlight the value of global periodic information while leaving open how spectral responses should change with the calendar context of a forecast. Recent methods also model amplitude and phase: APDNet uses real–imaginary spectral mappings (Zhuang et al., 2024); Implicit Forecaster predicts waves by frequency, amplitude, and phase (Li et al., 2025); FIRE separates amplitude and phase (He et al., 2025); and PAMNet uses cycle-aware amplitude and phase embeddings (Zhou et al., 2026). HarMoTC conditions channel- and frequency-specific amplitude rescaling and phase calibration on explicit calendar indices. These transformations can adapt both the strength and timing of recurring components as the calendar state changes.

Temporal encoding and exogenous information. The Temporal Fusion Transformer (TFT) (Lim et al., 2021) integrates temporal covariates through feature selection and attention; Time2Vec (Kazemi et al., 2019) learns periodic time representations. Such temporal signals can be supplied as auxiliary inputs or learned features of a predictor. HarMoTC instead uses Fourier calendar encodings to generate spectral modulation parameters, adapting the spectral transformation itself to temporal context. Consequently, similar histories can receive different spectral adjustments across calendar contexts.

3 METHOD

Figure 2 illustrates the overall architecture of HarMoTC. The shared forecasting backbone consists of instance normalization, Fourier transformation, adaptive spectral gating, frequency-wise cross-variable interaction, inverse transformation, and temporal projection. HarMoTC augments this backbone with two core components: calendar-conditioned spectral modulation and phase–amplitude decoupled spectral modeling.

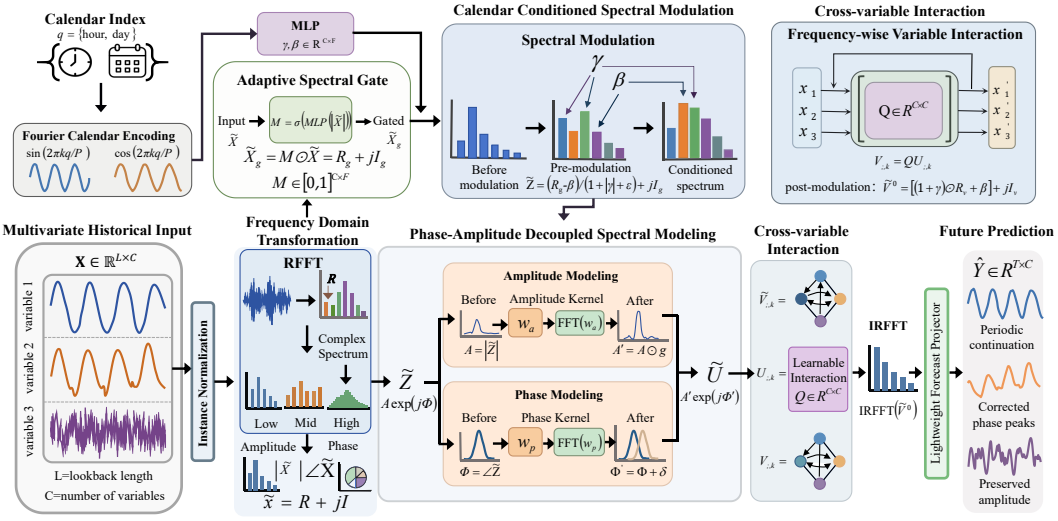

Figure 2: Overview of HarMoTC and its two proposed components: calendar-conditioned spectral modulation and phase–amplitude decoupled spectral modeling. The diagram shows the general configuration with hour-of-day and day-of-week conditioning; either can be disabled at run time.

3.1 Problem Definition and Overview

Long-term time-series forecasting predicts the next T time steps from a historical multivariate sequence of length L. Let C denote the number of variables. Given an input sequence $\mathbf{X} \in \mathbb{R}^{L\times C}$ and its calendar state $\mathbf{q} = (q^h, q^d)$, where q^h and q^d denote the hour-of-day and day-of-week indices at the forecast origin (the first timestamp after the input window), HarMoTC produces $\widehat{\mathbf{Y}} \in \mathbb{R}^{T\times C}$:

$$\widehat{\mathbf{Y}} = f_\theta(\mathbf{X}, \mathbf{q}). \tag{1}$$

HarMoTC first performs instance-wise normalization and transforms the normalized input into the frequency domain:

$$\widetilde{\mathbf{X}} = \mathrm{RFFT}(\overline{\mathbf{X}}) \in \mathbb{C}^{C\times F}, \qquad F = \left\lfloor \frac{L}{2} \right\rfloor + 1, \tag{2}$$

where $\overline{\mathbf{X}}$ is the normalized input and F is the number of non-redundant frequency components.

As part of the shared backbone, an adaptive spectral gate estimates the relevance of each frequency from the spectral magnitude:

$$\mathbf{M} = \sigma\Big(G_\xi\Big(|\widetilde{\mathbf{X}}|\Big)\Big) \in [0,1]^{C\times F}, \qquad \widetilde{\mathbf{X}}_g = \mathbf{M} \odot \widetilde{\mathbf{X}} = \mathbf{R}_g + \mathrm{j}\mathbf{I}_g, \tag{3}$$

where $G_\xi(\cdot)$ is a lightweight magnitude-based gating network and $\sigma(\cdot)$ is the sigmoid function. The same gating network is applied independently to the F-dimensional magnitude vector of each variable, producing a sample- and variable-specific mask $\mathbf{M}$. The gate suppresses less relevant frequencies while retaining the complex spectral representation. The batch dimension is omitted for clarity.

The gated spectrum is first mapped to a calendar-conditioned spectrum $\widetilde{\mathbf{Z}}$ through pre-modulation. Phase–amplitude modeling then transforms $\widetilde{\mathbf{Z}}$ into $\widetilde{\mathbf{U}}$. Frequency-wise cross-variable interaction is subsequently applied:

$$\widetilde{\mathbf{V}}_{:,k} = \mathbf{Q}\widetilde{\mathbf{U}}_{:,k}, \qquad k = 1, \ldots, F, \tag{4}$$

where $\mathbf{Q} \in \mathbb{R}^{C\times C}$ is a learnable interaction matrix shared across frequencies. The same real-valued transformation is applied to the real and imaginary components, enabling information exchange among variables while preserving the complex representation.

Calendar-conditioned post-modulation then maps $\widetilde{\mathbf{V}}$ to the final conditioned spectrum $\widetilde{\mathbf{V}}^o$. Finally, the spectrum is transformed back to the time domain and mapped to the prediction horizon by a lightweight temporal projector $P_\omega(\cdot)$:

$$\widehat{\mathbf{Y}} = \mathrm{Denorm}\left(P_\omega\left[\mathrm{IRFFT}\left(\widetilde{\mathbf{V}}^o; L\right)\right]\right). \tag{5}$$

3.2 Calendar-Conditioned Spectral Modulation

The first core component conditions the spectral transformation on the calendar state of each input window. Directly concatenating calendar features with observations does not explicitly characterize their effects on individual variables and frequencies. We therefore generate channel- and frequency-specific modulation parameters from periodic calendar indices.

For a calendar index q with period P, we construct a periodic representation using K Fourier harmonics:

$$\begin{aligned} \vartheta_{P,k}(q) &= \frac{2\pi k(q \bmod P)}{P}, \\ \boldsymbol{\psi}_P(q) &= \left[\sin\vartheta_{P,k}(q),\, \cos\vartheta_{P,k}(q)\right]_{k=1}^{K}. \end{aligned} \tag{6}$$

For the general case in which both calendar variables are enabled, the hour and day representations are concatenated and mapped to calendar-dependent modulation parameters:

$$\mathbf{e}_q = \left[\boldsymbol{\psi}_{P_h}(q^h); \boldsymbol{\psi}_{P_d}(q^d)\right], \qquad (\boldsymbol{\gamma}, \boldsymbol{\beta}) = G_\eta(\mathbf{e}_q), \tag{7}$$

where $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^{C\times F}$. Thus, each input window obtains modulation parameters that can adapt individual variables and frequencies to its temporal context. If only one calendar variable is enabled

by a run configuration, the other representation is omitted from $\mathbf{e}_q$; the remaining architecture is unchanged. The verified configuration details are given in Appendix B.1.

Using the gated spectrum obtained from Equation (3), written as $\widetilde{\mathbf{X}}_g = \mathbf{R}_g + \mathrm{j}\mathbf{I}_g$, HarMoTC performs calendar-conditioned pre-modulation before phase–amplitude modeling:

$$\widetilde{\mathbf{Z}} = \frac{\mathbf{R}_g - \boldsymbol{\beta}}{1 + |\boldsymbol{\gamma}| + \epsilon} + \mathrm{j}\mathbf{I}_g, \tag{8}$$

where ϵ is a small constant for numerical stability. This operation maps the gated spectrum into a calendar-conditioned representation space while retaining its imaginary component.

After phase–amplitude transformation and frequency-wise cross-variable interaction, write the resulting spectrum as $\widetilde{\mathbf{V}} = \mathbf{R}^v + \mathrm{j}\mathbf{I}^v$. The calendar-dependent parameters are reused to modulate this transformed spectrum:

$$\widetilde{\mathbf{V}}^o = [(1 + \boldsymbol{\gamma}) \odot \mathbf{R}^v + \boldsymbol{\beta}] + \mathrm{j}\mathbf{I}^v, \tag{9}$$

where $\odot$ denotes element-wise multiplication. The pre- and post-modulation operations form an asymmetric conditional transformation: pre-modulation maps the gated input spectrum into a conditioned representation space, whereas post-modulation adjusts the transformed spectrum using the same temporal context. Calendar information therefore directly controls variable- and frequency-specific spectral responses rather than serving only as an additional input feature.

3.3 Phase–Amplitude Decoupled Spectral Modeling

The second core component uses an explicit polar parameterization of the conditioned spectrum. Spectral amplitude characterizes the strength of individual periodic components, whereas spectral phase determines their relative harmonic alignment and consequently affects the reconstructed waveform. HarMoTC decomposes the calendar-conditioned spectrum as

$$\widetilde{\mathbf{Z}} = \mathbf{A} \odot \exp(\mathrm{j}\boldsymbol{\Phi}), \qquad \mathbf{A} = |\widetilde{\mathbf{Z}}|, \quad \boldsymbol{\Phi} = \angle\widetilde{\mathbf{Z}}. \tag{10}$$

Two independent learnable temporal kernels $\mathbf{w}_a, \mathbf{w}_p \in \mathbb{R}^L$ generate frequency-dependent amplitude and phase responses:

$$\mathbf{g} = |\mathrm{RFFT}(\mathbf{w}_a)|, \qquad \boldsymbol{\delta} = \angle\,\mathrm{RFFT}(\mathbf{w}_p). \tag{11}$$

The nonnegative response $\mathbf{g} \in \mathbb{R}^F_{\geq 0}$ controls the amplification or attenuation of individual frequency components, while $\boldsymbol{\delta} \in [-\pi, \pi]^F$ controls their relative phase alignment.

The transformed spectrum is reconstructed as

$$\widetilde{\mathbf{U}} = (\mathbf{A} \odot \mathbf{g}) \odot \exp[\mathrm{j}(\boldsymbol{\Phi} + \boldsymbol{\delta})]. \tag{12}$$

The amplitude response explicitly strengthens or attenuates periodic components, whereas the phase response adjusts their relative harmonic alignment. A linear phase response corresponds to a uniform temporal shift, while a general frequency-dependent response enables more flexible waveform adjustment. The two responses are shared across variables; calendar dependence is supplied by the modulation component, and cross-variable information exchange is provided by Equation (4).

3.4 Training Objective

HarMoTC and all architectural variants are trained using the same complex spectral discrepancy. For a batch of B samples, the ground-truth and predicted future sequences are transformed along the forecasting dimension:

$$\mathbf{S}^{(b)} = \mathcal{F}_T\left(\mathbf{Y}^{(b)}\right), \qquad \widehat{\mathbf{S}}^{(b)} = \mathcal{F}_T\left(\widehat{\mathbf{Y}}^{(b)}\right), \tag{13}$$

where $\mathcal{F}_T$ denotes the length-T discrete Fourier transform.

The model minimizes the average complex spectral discrepancy:

$$\mathcal{L}_{\mathrm{spec}} = \frac{1}{BTC} \sum_{b=1}^{B} \sum_{k=0}^{T-1} \sum_{c=1}^{C} \left| \widehat{S}^{(b)}_{k,c} - S^{(b)}_{k,c} \right|, \tag{14}$$

where $|\cdot|$ denotes the complex modulus. Unlike a magnitude-only objective, this loss compares complete complex coefficients and therefore penalizes discrepancies in both spectral magnitude and phase. All model parameters are optimized end-to-end using this common objective.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate HarMoTC on seven multivariate long-term forecasting benchmarks: ETTm1, ETTm2, ETTh1, ETTh2, Weather, Electricity, and Traffic (Zhou et al., 2021; Wu et al., 2021). They span different sampling granularities, dimensionalities, and application domains. Following the standard protocol, we fix the look-back length to $L = 96$ and use prediction horizons $T \in \{96, 192, 336, 720\}$. Each entry in Table 1 is averaged over these four horizons.

Baselines. We compare HarMoTC with eight representative forecasting methods: Transformer-based SSformer (Liu et al., 2026), iTransformer (Liu et al., 2024), and PatchTST (Nie et al., 2023); two-dimensional Times2D (Nematirad et al., 2025); retrieval-augmented RAFT (Han et al., 2025); frequency-domain FITS (Xu et al., 2024); MLP-based TimeMixer (Wang et al., 2024a); and linear DLinear (Zeng et al., 2023).

Implementation Details. We report mean squared error (MSE) and mean absolute error (MAE), where lower is better. All HarMoTC variants are trained end-to-end using Equation (14). HarMoTC is implemented in PyTorch 2.8.0+cu126 and trained on an NVIDIA A800 80GB GPU using Adam. Within each dataset–horizon setting, all variants use the same optimizer and training protocol. The HarMoTC entries in Table 1 are single-run values averaged over the four horizons. A separate three-seed robustness analysis is reported in Section 4.4 (Table 5). Further preprocessing, random-seed settings, optimization, checkpoint, and configuration details are provided in Appendix B.1.

4.2 MAIN RESULTS

Long-term Forecasting. Table 1 summarizes the long-term forecasting results on seven benchmarks. Overall, HarMoTC achieves the best or tied-best performance on 11 out of 14 dataset–metric pairs. Macro-averaging the seven dataset-level entries yields an MSE of 0.3221 and an MAE of 0.3344 for HarMoTC. The strongest baseline macro-average for both metrics is obtained by SSformer, with 0.3327 MSE and 0.3400 MAE. HarMoTC therefore reduces the macro-averaged MSE and MAE by 3.18% and 1.64%, respectively.

Table 1: Long-term forecasting results averaged over $T \in \{96, 192, 336, 720\}$ with $L = 96$. Lower is better; bold and underlined values denote the best and second-best results, respectively, with ties formatted identically.

Dataset	**HarMoTC (Ours)**		SSformer 2026		Times2D 2025		RAFT 2025		FITS 2024		iTransformer 2024		TimeMixer 2024		PatchTST 2023		DLinear 2023	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	**0.359**	**0.380**	0.380	0.394	0.392	0.403	0.381	0.400	0.415	0.408	0.407	0.409	0.381	0.396	0.387	0.400	0.403	0.407
ETTm2	**0.263**	**0.308**	0.279	0.322	0.278	0.326	0.281	0.330	0.286	0.328	0.288	0.332	0.275	0.323	0.281	0.326	0.350	0.401
ETTh1	**0.412**	**0.425**	0.452	0.443	0.433	0.430	0.428	0.433	0.451	0.440	0.454	0.448	0.447	0.440	0.469	0.455	0.456	0.452
ETTh2	**0.353**	**0.386**	0.359	0.389	0.364	0.393	0.382	0.410	0.383	0.408	0.383	0.406	0.365	0.395	0.387	0.407	0.559	0.515
Weather	**0.232**	**0.266**	0.237	0.268	0.261	0.281	0.270	0.309	0.249	0.276	0.258	0.278	0.240	0.272	0.259	0.280	0.265	0.317
Electricity	**0.170**	0.267	**0.170**	**0.265**	0.198	0.287	0.175	0.272	0.217	0.295	0.178	0.270	0.182	0.273	0.205	0.290	0.212	0.300
Traffic	0.466	0.309	0.452	0.299	0.530	0.356	**0.414**	0.284	0.627	0.376	0.428	**0.282**	0.485	0.297	0.481	0.304	0.625	0.383

HarMoTC is particularly effective on the four ETT benchmarks, ranking first on all eight ETT dataset–metric pairs. Averaged over ETTm1, ETTm2, ETTh1, and ETTh2, HarMoTC reaches 0.3468 MSE and 0.3748 MAE. The strongest competing ETT averages are 0.3668 MSE from Times2D and 0.3870 MAE from SSformer, corresponding to relative reductions of 5.45% and 3.17%, respectively. These consistent gains across both hourly and minute-level ETT subsets indicate that the proposed method captures temporal patterns effectively at different sampling granularities.

On Weather, HarMoTC achieves the best MSE and MAE of 0.232 and 0.266, improving over the second-best SSformer results by 0.005 and 0.002, respectively. On Electricity, HarMoTC ties SSformer for the best MSE of 0.170 and obtains the second-best MAE of 0.267, compared with

0.265 for SSformer. The results on Traffic are comparatively less competitive: RAFT obtains the best MSE of 0.414, while iTransformer achieves the best MAE of 0.282, compared with $0.466/0.309$ for HarMoTC. Traffic therefore remains a challenging setting for the current model and provides a clear direction for further improvement. Taken together, the results demonstrate strong performance across the ETT, Weather, and Electricity benchmarks while also identifying a dataset on which further improvement is needed.

Ablation Study. To quantify the contributions and complementarity of the two core designs, we conduct ablation experiments on ETTm1, ETTh2, Weather, and Electricity under the same forecasting protocol as the main experiments. The w/o TC variant removes calendar-conditioned spectral modulation while retaining phase–amplitude modeling. The w/o Amp&Phase variant retains temporal conditioning but removes the complete phase–amplitude transformation, whereas w/o Amp and w/o Phase isolate the roles of its two branches. The spectral gate, linear channel-mixing layer, temporal projector, and training objective are kept unchanged across all variants. As shown in Table 2, the complete HarMoTC model achieves the best result on all eight dataset–metric pairs, supporting the effectiveness of both core designs.

Table 2: Core-design ablation averaged over $T \in \{96, 192, 336, 720\}$ with $L = 96$. "Avg." is the macro-average across the four datasets. Lower values are better; the best and second-best results are shown in bold and underlined, respectively, and ties receive the same formatting.

	MSE					MAE				
Configuration	ETTm1	ETTh2	Weather	Elec.	Avg.	ETTm1	ETTh2	Weather	Elec.	Avg.
HarMoTC	**0.359**	**0.353**	**0.232**	**0.170**	**0.279**	**0.380**	**0.386**	**0.266**	**0.267**	**0.325**
w/o TC	0.381	0.367	0.248	0.191	0.297	0.394	0.394	0.279	0.284	0.338
w/o Amp	0.366	0.363	0.241	0.178	0.287	0.386	<u>0.391</u>	0.273	0.271	0.330
w/o Phase	<u>0.364</u>	<u>0.358</u>	<u>0.238</u>	<u>0.177</u>	<u>0.284</u>	<u>0.384</u>	<u>0.391</u>	<u>0.272</u>	<u>0.270</u>	<u>0.329</u>
w/o Amp&Phase	0.371	0.365	0.244	0.183	0.291	0.391	0.393	0.276	0.279	0.335

Among the variants, removing temporal conditioning causes the largest degradation on every dataset. Using the unrounded macro-averages, MSE and MAE increase from $0.2785/0.3248$ to $0.2968/0.3378$, corresponding to relative increases of 6.55% and 4.00%, respectively. This result shows that calendar-driven temporal conditioning makes the largest empirical contribution among the evaluated components. Removing the amplitude or phase branch individually increases the average MSE/MAE to $0.2870/0.3303$ and $0.2843/0.3293$, respectively. The lower errors of w/o Phase indicate that retaining the amplitude branch is more beneficial on average, while its consistent gap from the complete model confirms the additional value of phase calibration. When both spectral branches are removed, the average errors rise to $0.2908/0.3348$, exceeding either single-branch ablation and supporting the complementary roles of amplitude and phase modeling.

4.3 Model Analysis

Visualization of Learned Spectral Transformations. To examine the internal behavior of HarMoTC, we visualize the spectral responses learned from ETTm1 using the best validation checkpoint with input length $L = 96$.

For Figure 3(a), the channel-aggregated modulation intensity is robustly centered and scaled over the 24 hourly states at each frequency; values are clipped to $[-2.5, 2.5]$ only for visualization. Each cell therefore represents one discrete hour–frequency pair, and its color describes variation within the same frequency rather than absolute differences between frequencies. The resulting pattern varies jointly across hours and temporal periods, indicating that HarMoTC does not apply a fixed spectral transformation to every temporal context.

Figures 3(b) and 3(c) further show that the two spectral branches learn distinct frequency-dependent responses. Relative to its median-normalized reference, the amplitude branch assigns larger responses to some frequency components and smaller responses to others, whereas the phase branch produces both positive and negative phase displacements relative to the zero-shift reference. These visualizations provide an interpretable view of how HarMoTC performs hour-conditioned spectral adaptation

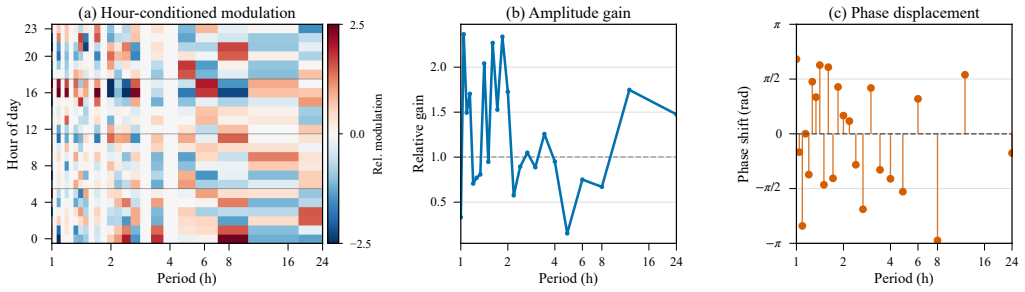

Figure 3: Learned spectral transformations on ETTm1: (a) hour-conditioned relative modulation, (b) median-normalized amplitude response, and (c) frequency-dependent phase displacement. Dashed lines mark the median reference and zero phase shift in (b) and (c), respectively.

and separately adjusts periodic strength and temporal alignment, while the corresponding forecasting benefits are established quantitatively by the core-design ablation results in Table 2.

Calendar-Stratified Spectral Fidelity. To test whether calendar conditioning improves spectral structure beyond aggregate error, we group ETTm1 test windows by forecast-origin hour at each horizon (96 hour–horizon groups) and measure normalized magnitude error and target-amplitude-weighted circular phase error over non-DC frequencies.

Table 3 shows reductions of 1.65% and 5.18%, with HarMoTC winning 77/96 amplitude groups and all 96 phase groups. The all-state phase gain and broad amplitude gain provide direct evidence that temporal conditioning aligns forecast spectra with calendar-dependent dynamics.

Table 3: Calendar-stratified spectral errors on ETTm1; lower is better. "Wins" counts hour–horizon groups improved by HarMoTC.

Metric	HarMoTC	w/o TC	Red. (%)	Wins
Amplitude error	0.6710	0.6823	+1.65	77/96
Phase error	0.5692	0.6003	+5.18	96/96

Counterfactual Calendar Alignment. To test whether HarMoTC functionally uses the semantic alignment of its calendar condition, we conduct an inference-only intervention on ETTm1. For each horizon $T \in \{96, 192, 336, 720\}$, we load the fixed best-validation checkpoint and keep the input series, targets, normalization, sample order, and model parameters unchanged. We modify only the forecast-origin hour-of-day index, using bidirectional shifts of ± 3 and ± 6 hours, a 12-hour shift, and five full-test-set random permutations. The two directions and five permutations are averaged within their respective conditions. The results reported in Table 4 are averaged over the four prediction horizons $T \in \{96, 192, 336, 720\}$.

Table 4 shows that correct alignment is consistently optimal. Relative MSE increases by 31.21%, 79.91%, and 137.91% under the ± 3-hour, ± 6-hour, and 12-hour shifts, respectively, while shuffled hour indices yield a 72.02% increase. The amplitude and phase errors also increase with deterministic displacement, confirming reliance on correctly aligned calendar semantics.

Table 4: Counterfactual calendar-alignment results on ETTm1 across four horizons.

Calendar input	MSE	ΔMSE (%)	Amp.	Phase
Correct	**0.3597**	**+0.00**	**0.6666**	**0.9842**
± 3 h	0.4706	+31.21	0.7041	1.1446
± 6 h	0.6429	+79.91	0.7215	1.2751
12 h	0.8449	+137.91	0.7557	1.4703
Shuffled	0.6141	+72.02	0.7168	1.2426

Paired circular moving-block bootstrap with block length T for overlapping windows gives intervals of $[28.41, 34.40]\%$, $[73.01, 87.63]\%$, $[124.60, 153.61]\%$, and $[65.83, 78.98]\%$, respectively; all exclude zero. The MSE ordering holds at every horizon (Appendix Table 9). Spectral errors in Table 4 use global test-set energy normalization and are not directly comparable with the calendar-group errors in Table 3. Because only calendar indices change, the degradation indicates that correct alignment matters for spectral adaptation.

4.4 Efficiency and Robustness

Efficiency.
We compare HarMoTC with eight representative baselines on ETTm1 under the $96 \rightarrow 96$ setting, using the same hardware and experimental protocol. In Figure 4, lower MSE and training time are preferable, while bubble size encodes parameter count. HarMoTC requires 35.6 seconds per training epoch, which is 21.1% faster than iTransformer and 39.0% faster than PatchTST. The differences become more pronounced relative to Times2D, RAFT, and SSformer.

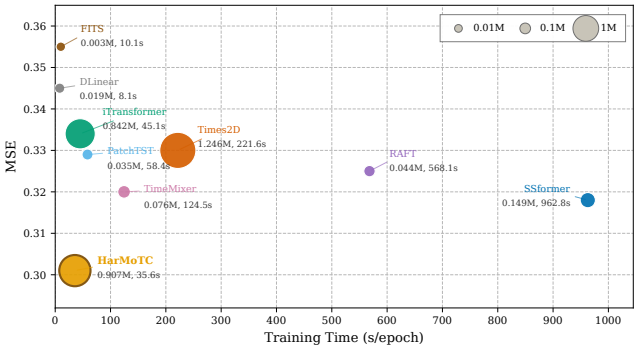

Figure 4: ETTm1 accuracy–efficiency trade-off under the $96 \rightarrow 96$ setting.

Their training times range from 221.6 to 962.8 seconds per epoch. Although FITS and DLinear remain faster and smaller, they produce MSEs of 0.355 and 0.345, respectively, compared with 0.301 for HarMoTC. HarMoTC thus attains the lowest error with moderate computational cost and fewer than one million parameters.

Robustness to Random Seeds.
Table 5 reports results from three independent HarMoTC runs. Across all seven datasets, the absolute standard deviations are at most 0.003 for both metrics, while the relative standard deviation (RSD) remains below 1.30% for MSE and 1.00% for MAE. These small variations indicate stable performance across changes in initialization and stochastic optimization.

Table 5: Random-seed robustness of HarMoTC over three runs. Values are mean ± standard deviation; RSD (%) is in parentheses.

Dataset	MSE (RSD)	MAE (RSD)
ETTh1	0.413 ± 0.001 (0.24)	0.425 ± 0.001 (0.24)
ETTh2	0.353 ± 0.002 (0.57)	0.386 ± 0.001 (0.26)
ETTm1	0.359 ± 0.002 (0.56)	0.380 ± 0.001 (0.26)
ETTm2	0.263 ± 0.001 (0.38)	0.308 ± 0.001 (0.32)
Weather	0.232 ± 0.003 (1.29)	0.266 ± 0.002 (0.75)
Traffic	0.466 ± 0.003 (0.64)	0.309 ± 0.003 (0.97)
Electricity	0.170 ± 0.002 (1.18)	0.265 ± 0.002 (0.75)

4.5 Limitations and Future Work

HarMoTC is less competitive on Traffic, where its static, frequency-shared cross-variable interaction may be insufficient for heterogeneous dependencies; this explanation requires controlled comparisons with alternative interactions. Moreover, calendar indices cannot distinguish different regimes that share the same hour or day. Our evaluation uses seven regularly sampled benchmarks with a fixed look-back length $L = 96$, so generalization to irregular inputs and longer contexts remains untested. Future work will evaluate dynamic or sparse interactions and assess robustness to unobserved events, irregular sampling, and alternative context lengths. Appendix A discusses these boundaries in detail.

5 Conclusion

We propose HarMoTC, a framework that performs temporally conditioned phase–amplitude spectral modeling for multivariate time-series forecasting. Calendar-conditioned spectral modulation uses multi-harmonic Fourier encodings to adapt spectral transformations to temporal context, while phase–amplitude decoupled modeling separately reconstructs periodic strength and temporal alignment. The remaining operations form a shared spectral forecasting backbone rather than additional methodological contributions. Across seven long-term forecasting benchmarks, HarMoTC achieves the best or tied-best values on 11 of 14 dataset–metric pairs in Table 1. The ablation study, learned-response visualizations, and counterfactual calendar-alignment analysis further support the effectiveness and complementarity of the two proposed designs.

AI USE STATEMENT

Generative AI tools were used solely for manuscript translation and language polishing. All AI-assisted text was reviewed and edited by the authors, who take full responsibility for the final content of this paper.

ETHICS STATEMENT

This work studies forecasting algorithms on established public time-series benchmarks and does not involve human subjects or the collection of personally identifiable information. Deployment in high-impact operational settings should include domain-specific validation and human oversight because inaccurate forecasts may affect downstream decisions.

REPRODUCIBILITY STATEMENT

The experimental setup, optimization details, data preprocessing, random-seed protocol, complete per-horizon results, and architectural boundaries are reported in the main paper and Appendices A and B. The submission package should include the source code and launch configurations used to reproduce the reported experiments.

REFERENCES

Cristian Challu, Kin G. Olivares, Boris N. Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. NHITS: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023. doi: 10.1609/aaai.v37i6.25854.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan O. Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21774–21797, 2025.

Cheng He, Xijie Liang, Zengrong Zheng, Patrick P. C. Lee, Xu Huang, Zhaoyi Li, Hong Xie, Defu Lian, and Enhong Chen. A unified frequency domain decomposition framework for interpretable and robust time series forecasting. arXiv preprint arXiv:2510.10145, 2025. URL https://arxiv.org/abs/2510.10145.

Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus A. Brubaker. Time2Vec: Learning a vector representation of time. *CoRR*, abs/1907.05321, 2019. URL https://arxiv.org/abs/1907.05321.

Xinyu Li, Yuchen Luo, Hao Wang, Haoxuan Li, Liuhua Peng, Feng Liu, Yandong Guo, Kun Zhang, and Mingming Gong. Towards accurate time series forecasting via implicit decoding. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/0e82ef0c89df6a6eff8734ea7e27c42f-Abstract-Conference.html.

Bryan Lim, Sercan O. Arik, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4): 1748–1764, 2021. doi: 10.1016/j.ijforecast.2021.03.012.

Ying Liu, Bo Liu, Sheng Huang, Gang Luo, Wenbo Hu, Meng Wang, and Richang Hong. Sparse-scale transformer with bidirectional awareness for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 23899–23907, 2026. doi: 10.1609/aaai.v40i28.39566.

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 *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

Reza Nematirad, Anil Pahwa, and Balasubramaniam Natarajan. Times2D: Multi-period decomposition and derivative mapping for general time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19651–19658, 2025. doi: 10.1609/aaai.v39i18.34164.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Hao Wang, Lichen Pan, Yuan Shen, Zhichao Chen, Degui Yang, Yifei Yang, Sen Zhang, Xinggao Liu, Haoxuan Li, and Dacheng Tao. FreDF: Learning to forecast in the frequency domain. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1457fb1e5d72cdc4ecd88bc10f916095-Abstract-Conference.html.

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 *International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=7oLshfEIC2.

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. In *Advances in Neural Information Processing Systems*, volume 37, 2024b. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0113ef4642264adc2e6924a3cbbdf532-Abstract-Conference.html.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, volume 34, pp. 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 *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=bWcnvZ3qMb.

Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Defu Lian, Ning An, Longbing Cao, and Zhendong Niu. Frequency-domain MLPs are more effective learners in time series forecasting. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

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, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

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, pp. 11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286, 2022.

Yingbo Zhou, Yutong Ye, Zhiwei Ling, Shuhao Li, Rui Qian, Jian Xiong, Li Sun, and Dejing Dou. PAMNet: Cycle-aware phase-amplitude modulation network for multivariate time series forecasting. arXiv preprint arXiv:2605.02938, 2026. URL https://arxiv.org/abs/2605.02938.

Wei Zhuang, Jili Fan, Jiayu Fang, Wenxuan Fang, and Min Xia. Rethinking general time series analysis from a frequency domain perspective. *Knowledge-Based Systems*, 301:112281, 2024. doi: 10.1016/j.knosys.2024.112281.

A Failure Modes and Boundary Analysis

This appendix clarifies the operating conditions under which HarMoTC is expected to be effective and identifies cases in which its current design may be insufficient. We distinguish an *observed failure case*, directly supported by the reported experiments, from *architectural boundaries* that follow from the model formulation but have not all been independently verified by dedicated stress tests. HarMoTC is primarily designed for regularly sampled multivariate series whose future dynamics contain repeatable spectral structure and whose regime variations are meaningfully indexed by calendar variables.

Table 6: Main architectural boundaries of HarMoTC. The Traffic result is observed empirically; the remaining rows are expected failure modes implied by the current formulation and require dedicated stress tests.

Boundary condition	Mechanism	Expected symptom	Recommended diagnostic
Highly heterogeneous variable relations	Static, dense $\mathbf{Q}$ shared across samples and frequencies	Weak performance on localized or topology-dependent dynamics	Compare dynamic, graph-based, sparse, and frequency-specific interactions
Off-grid, long, or drifting periods	Finite look-back window and discrete RFFT grid	Spectral leakage, blurred peaks, and horizon-dependent phase error	Sweep L and inject controlled off-grid or chirped signals
Unobserved temporal regimes	Conditioning uses only the enabled forecast-origin calendar indices	Similar predictions for calendar-identical but behaviorally different states	Stratify holidays, incidents, interventions, and timestamp corruption
Variable-specific spectral dynamics	Amplitude and phase responses are shared across variables	Under-correction for variables with distinct delays or periodicities	Compare shared, grouped, and variable-specific kernels
Missing or irregular observations	FFT assumes a uniformly sampled complete window	Artificial frequencies and unstable reconstruction	Vary missing rate, gap length, and sampling irregularity
Very large C	Dense frequency-wise $C \times C$ interaction	Quadratic growth in compute and memory	Report scaling curves in C and test sparse or low-rank alternatives

A.1 Observed Failure Case: Heterogeneous Traffic Dynamics

Traffic is the clearest empirical failure case. HarMoTC reports $0.466/0.309$ MSE/MAE, versus the best reported 0.414 MSE (RAFT) and 0.282 MAE (iTransformer). Relative to these baselines, MSE and MAE are approximately 12.6% and 9.6% higher, respectively, contrasting with strong results on ETT, Weather, and Electricity.

A plausible explanation is HarMoTC's cross-variable interaction. At every frequency k, the model applies the same learnable matrix $\mathbf{Q} \in \mathbb{R}^{C \times C}$,

$$\widetilde{\mathbf{V}}_{:,k} = \mathbf{Q}\widetilde{\mathbf{U}}_{:,k}. \tag{15}$$

This operation enables global exchange but is static across samples and shared across frequencies. Traffic networks may contain location-specific dependencies, localized disruptions, and frequency-dependent relations that cannot be adequately represented by one global matrix. The Traffic result suggests a limitation under highly heterogeneous cross-variable structure. This explanation is consistent with the architecture, but a causal conclusion would require a dedicated comparison with dynamic, sparse, graph-structured, or frequency-specific interaction modules.

A.2 Spectral Resolution and Nonstationary Periodicity

HarMoTC uses the discrete RFFT grid of a length-L input window, imposing three limits. First, periods much longer than the look-back window are hard to identify from one window. Second, off-grid frequencies may spread energy across adjacent bins, reducing the precision of frequency-wise

rescaling and phase calibration. Third, rapidly drifting periods preclude a stable Fourier summary of the window.

Abrupt spikes, incidents, structural breaks, and other nonperiodic transients may also spread energy over many frequency components. The adaptive spectral gate can suppress weak components, but it does not explicitly separate a persistent oscillation from a short-lived event with a broad spectrum. HarMoTC may preserve the periodic background while smoothing, delaying, or misestimating rare transients.

A.3 Limits of Calendar Conditioning

The temporal-conditioning module can encode forecast-origin hour-of-day and day-of-week through fixed-period Fourier harmonics. The implementation exposes the two calendar variables as the independent `use_hour_index` and `use_day_index` configuration options. The main paper shows both enabled; disabling either omits its Fourier representation from the conditioning vector without changing the forecasting backbone. Verified ETTh1 configurations enable hour-of-day and disable day-of-week conditioning. The module can exploit regimes recurring with the enabled calendar state but cannot distinguish different situations sharing that state. Examples include public holidays, daylight-saving transitions, equipment maintenance, weather emergencies, traffic incidents, policy interventions, and other exogenous regime changes. Incorrect or shifted timestamps may be especially harmful because they directly generate inappropriate modulation parameters rather than merely adding a noisy auxiliary feature.

Given a gated spectrum $\widetilde{\mathbf{X}}_g = \mathbf{R}_g + j\mathbf{I}_g$, HarMoTC applies calendar-dependent pre- and post-modulation to the real component,

$$\widetilde{\mathbf{Z}} = \frac{\mathbf{R}_g - \boldsymbol{\beta}}{1 + |\boldsymbol{\gamma}| + \epsilon} + j\mathbf{I}_g, \tag{16}$$

$$\widetilde{\mathbf{V}}^o = \left[(1 + \boldsymbol{\gamma}) \odot \mathbf{R}_v + \boldsymbol{\beta}\right] + j\mathbf{I}_v. \tag{17}$$

Thus, calendar context does not independently modulate the real and imaginary components. Context-dependent phase changes can still arise indirectly after polar decomposition and phase calibration, but the conditioning mechanism itself is not a fully general complex-valued transformation. This may restrict adaptation when temporal context primarily changes phase rather than spectral strength.

A.4 Shared Phase–Amplitude Responses

The amplitude gain and phase displacement are generated from two learnable temporal kernels and are shared across variables:

$$\mathbf{g} = |\,\mathrm{RFFT}(\mathbf{w}_a)|, \qquad \boldsymbol{\delta} = \angle\,\mathrm{RFFT}(\mathbf{w}_p). \tag{18}$$

Sharing reduces parameters and provides a common spectral inductive bias, but it also assumes that variables benefit from the same frequency-wise gain and phase response. This assumption may be restrictive when variables have different sensor dynamics, delays, dominant periods, or noise levels. Calendar modulation supplies variable-specific parameters, but it does not fully replace variable-specific amplitude and phase kernels.

Phase is weakly informative near zero amplitude: large phase shifts may barely affect reconstruction, while small perturbations can destabilize phase values. HarMoTC's complete complex spectral objective reduces the risk of optimizing phase independently of magnitude, but the phase branch may still receive limited useful information at low-energy frequencies.

A.5 Data and Output Boundaries

HarMoTC assumes complete, uniformly sampled input windows. Missing observations, irregular sampling intervals, asynchronous variables, and changes in sampling frequency are not explicitly modeled. FFT on naively imputed or irregularly sampled data can introduce artificial spectral components, requiring preprocessing or model extensions.

HarMoTC produces deterministic point forecasts without modeling uncertainty or multiple plausible futures. This is a relevant boundary in safety-critical or decision-making applications where calibrated

intervals, tail risks, or scenario diversity are required. Moreover, the reported experiments use a fixed look-back length $L = 96$ and seven standard benchmarks. The current evidence therefore does not establish robustness to substantially different context lengths, sampling granularities, domains, or large distribution shifts.

A.6 Computational Boundary

For C variables and F nonredundant frequency bins, the dense cross-variable multiplication requires $\mathcal{O}(FC^2)$ operations per sample and stores a $C \times C$ interaction matrix. The calendar module also generates modulation tensors of size $C \times F$. The reported wall-clock results show favorable efficiency on the evaluated benchmarks, but the dense interaction can become a bottleneck for datasets with thousands of variables. Sparse, low-rank, grouped, or graph-constrained interaction would be more appropriate in that regime.

B Additional Experimental Details

B.1 Implementation and Reproducibility Details

Baseline Results. For each baseline, we use the results reported in its original paper when they match our evaluation setting. Otherwise, we obtain the results using the corresponding official implementation.

Software, hardware, and precision. HarMoTC is implemented in PyTorch 2.8.0+cu126 and was trained on an NVIDIA A800 GPU with 80 GB of memory. The supplied launch configurations do not enable automatic mixed precision, so model training and evaluation use FP32. The data loaders use zero worker processes in the verified configuration.

Randomization and number of runs. The implementation sets the Python `random`, NumPy, and PyTorch random seeds before model construction and data loading. The verified ETTh1 main-run configurations use one run with seed 2024. The random-seed robustness experiment repeats training with seeds 2024, 2025, and 2026 and reports the mean and standard deviation over the three runs.

Data preprocessing and window construction. All reported forecasting experiments use the multivariate-to-multivariate setting and a look-back length of $L = 96$. Each variable is standardized using a `StandardScaler` fitted only on the training portion, and the fitted statistics are then applied to the validation and test portions. The ETT loaders use the conventional chronological 12/4/4-month train/validation/test partition, while the Weather and generic custom loaders use chronological 70/10/20 percent partitions. Sliding windows are constructed without imputation or irregular-sampling correction.

Table 7: Dataset statistics. Split sizes give the numbers of time points in the train/validation/test partitions.

Dataset	Variables	Time-point split	Frequency
ETTh1, ETTh2	7	$(8640, 2880, 2880)$	Hourly
ETTm1, ETTm2	7	$(34560, 11520, 11520)$	15 min
Weather	21	$(36887, 5270, 10539)$	10 min
Electricity	321	$(18412, 2632, 5260)$	Hourly
Traffic	862	$(12280, 1756, 3508)$	Hourly

In addition to dataset-level standardization, HarMoTC applies instance-wise normalization independently to each sample and variable over the look-back dimension,

$$\bar{\mathbf{X}} = \frac{\mathbf{X} - \boldsymbol{\mu}}{\sqrt{\boldsymbol{\sigma}^2 + 10^{-5}}}, \tag{19}$$

and reverses this transformation after the temporal projection. The calendar index passed to the model is taken at the forecast origin, i.e., the first timestamp immediately following the input window. In the verified ETTh1 configuration, hour-of-day conditioning with period 24 is enabled and day-of-week conditioning is disabled.

Optimization, objective, and checkpoint selection. Models are optimized with Adam. Unless otherwise stated in Table 8, all unspecified Adam parameters use the PyTorch defaults. Batch size and the maximum number of training epochs vary across datasets and horizons; early stopping uses patience 5. The default `type3` learning-rate schedule keeps the initial learning rate through epoch 3 and then multiplies it by 0.8 after each subsequent epoch:

$$\eta_e = \begin{cases} \eta_0, & e \le 3, \\ \eta_0\, 0.8^{e-3}, & e > 3. \end{cases} \tag{20}$$

The checkpoint with the lowest validation time-domain MSE is saved and reloaded for final test evaluation. HarMoTC itself is trained only with the complete complex spectral MAE,

$$\mathcal{L}_{\text{spec}} = \text{mean}\Big(\Big|\text{FFT}_T(\widehat{\mathbf{Y}}) - \text{FFT}_T(\mathbf{Y})\Big|\Big), \tag{21}$$

where the FFT is taken along the prediction dimension and the complex modulus is applied before averaging. No additional time-domain term is mixed into the training objective. Final MSE and MAE are computed on the standardized test sequences.

Table 8: HarMoTC hyperparameters across seven datasets ($L = 96$, seed 2024).

Dataset	Hyperparameter	96	192	336	720
ETTh1	Learning rate	4×10^{-3}	3.5×10^{-3}	3.4×10^{-3}	1.7×10^{-3}
	Weight decay	2×10^{-4}	2×10^{-4}	3.2×10^{-4}	2×10^{-6}
	FiLM hidden width	256	256	512	128
	Fourier harmonics K	4	4	2	2
ETTh2	Learning rate	5.5×10^{-3}	4×10^{-3}	5.6×10^{-3}	5.6×10^{-3}
	Weight decay	3×10^{-5}	3×10^{-4}	4×10^{-4}	4×10^{-4}
	FiLM hidden width	128	128	128	128
	Fourier harmonics K	5	5	5	5
ETTm1	Learning rate	6×10^{-3}	9.3×10^{-3}	7×10^{-3}	7×10^{-3}
	Weight decay	1.5×10^{-3}	1×10^{-4}	5×10^{-4}	5×10^{-4}
	FiLM hidden width	32	256	64	64
	Fourier harmonics K	8	1	4	3
ETTm2	Learning rate	7.5×10^{-4}	7.5×10^{-4}	7.4×10^{-4}	7.5×10^{-4}
	Weight decay	6×10^{-5}	6×10^{-5}	6×10^{-5}	6×10^{-5}
	FiLM hidden width	64	64	64	128
	Fourier harmonics K	8	3	8	7
Traffic	Learning rate	1.2×10^{-4}	1.1×10^{-4}	1.1×10^{-4}	1.1×10^{-4}
	Weight decay	2.4×10^{-3}	2.5×10^{-3}	2.5×10^{-3}	2.5×10^{-3}
	FiLM hidden width	32	256	256	256
	Fourier harmonics K	2	4	4	4
Weather	Learning rate	7.4×10^{-4}	7.4×10^{-4}	2×10^{-4}	2×10^{-4}
	Weight decay	5×10^{-7}	5×10^{-7}	4.5×10^{-7}	3×10^{-7}
	FiLM hidden width	64	64	32	32
	Fourier harmonics K	1	1	7	2
Electricity	Learning rate	9.5×10^{-4}	1×10^{-3}	9.5×10^{-4}	9.5×10^{-4}
	Weight decay	4×10^{-7}	1×10^{-7}	4×10^{-7}	4×10^{-7}
	FiLM hidden width	32	32	128	128
	Fourier harmonics K	2	1	8	2

Verified model instantiation. For $L = 96$, an orthonormal RFFT yields $F = 49$ bins. A shared $49 \to 12 \to 49$ GELU–sigmoid MLP maps each variable's magnitude vector to sample- and variable-specific masks. For ETTh1 $96 \to 96$, eight hour-of-day Fourier features feed an $8 \to 256 \to 256$

GELU MLP with separate zero-initialized heads for $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^{C \times F}$. Amplitude and phase kernels are initialized as $0.02\mathcal{N}(0,1)$ and $\mathcal{N}(0, 0.01)$; the $C \times C$ interaction matrix starts from identity. Frequency dropout is 0.05. The projector is $96 \to 720 \to 720 \to T$ with LayerNorm, ReLU, and dropout 0.1 after each hidden layer. The verified checkpoint has 906,932 trainable parameters; the package includes its ETTh1 launch configuration.

B.2 Per-Horizon Counterfactual Calendar Alignment

Table 9 reports per-horizon results underlying the macro-average in Table 4. Every condition uses the same chronologically ordered test windows. Confidence intervals use a paired circular moving-block bootstrap with 2,000 repeats and block length equal to the horizon. Spectral errors sum their non-DC numerators and target spectral energy over the test set before normalization.

Table 9: Per-horizon counterfactual calendar alignment on ETTm1. ΔMSE and its 95% confidence interval are relative to Correct within the same horizon; lower is better.

T	Calendar input	MSE	ΔMSE (%)	95% CI	Amp. error	Phase error
96	Correct	0.3015	+0.00	–	0.6456	0.8961
96	±3 h	0.4271	+41.62	$[34.73, 49.53]$	0.6850	1.0870
96	±6 h	0.6195	+105.44	$[88.98, 125.29]$	0.6985	1.2134
96	12 h	0.9168	+204.00	$[167.90, 244.46]$	0.7275	1.4173
96	Shuffled	0.5979	+98.27	$[82.29, 116.40]$	0.6925	1.1742
192	Correct	0.3430	+0.00	–	0.7099	1.0620
192	±3 h	0.4234	+23.42	$[20.12, 27.73]$	0.7342	1.1752
192	±6 h	0.5830	+69.94	$[59.45, 82.75]$	0.7393	1.2920
192	12 h	0.7053	+105.61	$[87.88, 128.71]$	0.7666	1.3580
192	Shuffled	0.5572	+62.44	$[52.31, 75.34]$	0.7388	1.2510
336	Correct	0.3709	+0.00	–	0.6235	0.9304
336	±3 h	0.4813	+29.75	$[24.88, 35.19]$	0.6702	1.0885
336	±6 h	0.6234	+68.05	$[58.09, 80.63]$	0.7028	1.2181
336	12 h	0.8066	+117.44	$[95.48, 143.93]$	0.7563	1.4848
336	Shuffled	0.5994	+61.57	$[51.42, 74.05]$	0.6963	1.2021
720	Correct	0.4233	+0.00	–	0.6874	1.0482
720	±3 h	0.5506	+30.06	$[24.22, 36.75]$	0.7269	1.2274
720	±6 h	0.7459	+76.19	$[61.21, 92.48]$	0.7453	1.3769
720	12 h	0.9507	+124.57	$[95.85, 155.67]$	0.7724	1.6210
720	Shuffled	0.7020	+65.82	$[52.50, 80.57]$	0.7398	1.3431

Table 10: Complete per-horizon forecasting results with $L = 96$. "Avg." matches the main-paper comparison; lower is better, and bold/underline denote the best/second-best results.

Dataset	Horizon	**HarMoTC** (Ours)		SSformer (2026)		Times2D (2025)		RAFT (2025)		FITS (2024)		iTransformer (2024)		TimeMixer (2024)		PatchTST (2023)		DLinear (2023)	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.301**	**0.341**	0.318	0.355	0.330	0.364	0.325	0.369	0.355	0.375	0.334	0.368	0.320	0.357	0.329	0.367	0.345	0.372
	192	**0.343**	**0.369**	0.360	0.379	0.361	0.382	0.364	0.388	0.392	0.393	0.377	0.391	0.361	0.381	0.367	0.385	0.380	0.389
	336	**0.370**	**0.388**	0.393	0.402	0.407	0.417	0.391	0.407	0.424	0.414	0.426	0.420	0.390	0.404	0.399	0.410	0.413	0.413
	720	**0.423**	**0.423**	0.449	0.440	0.472	0.450	0.444	0.438	0.487	0.449	0.491	0.459	0.454	0.441	0.454	0.439	0.474	0.453
	Avg.	**0.359**	**0.380**	0.380	0.394	0.392	0.403	0.381	0.400	0.415	0.408	0.407	0.409	0.381	0.396	0.387	0.400	0.403	0.407
ETTm2	96	**0.164**	**0.243**	0.171	0.254	0.174	0.262	0.176	0.264	0.183	0.266	0.180	0.264	0.175	0.258	0.175	0.259	0.193	0.292
	192	**0.227**	**0.285**	0.235	0.302	0.241	0.303	0.241	0.306	0.247	0.305	0.250	0.309	0.237	0.299	0.241	0.302	0.284	0.362
	336	**0.282**	**0.322**	0.307	0.338	0.301	0.342	0.302	0.345	0.307	0.342	0.311	0.348	0.298	0.340	0.305	0.343	0.369	0.427
	720	**0.378**	**0.380**	0.403	0.395	0.397	0.396	0.404	0.405	0.407	0.399	0.412	0.407	0.391	0.396	0.402	0.400	0.554	0.522
	Avg.	**0.263**	**0.308**	0.279	0.322	0.278	0.326	0.281	0.330	0.286	0.328	0.288	0.332	0.275	0.323	0.281	0.326	0.350	0.401
ETTh1	96	**0.352**	**0.384**	0.373	0.393	0.376	0.395	0.376	0.401	0.386	0.396	0.386	0.405	0.375	0.400	0.414	0.419	0.386	0.400
	192	**0.400**	**0.416**	0.432	0.431	0.435	0.423	0.417	0.424	0.436	0.423	0.441	0.436	0.429	0.421	0.460	0.445	0.437	0.432
	336	**0.439**	**0.439**	0.483	0.454	0.457	**0.439**	0.454	0.441	0.478	0.444	0.487	0.458	0.484	0.458	0.501	0.466	0.481	0.459
	720	**0.458**	**0.463**	0.519	0.495	0.465	0.464	0.465	0.465	0.502	0.495	0.503	0.491	0.498	0.482	0.500	0.488	0.519	0.516
	Avg.	**0.412**	**0.425**	0.452	0.443	0.433	0.430	0.428	0.433	0.451	0.440	0.454	0.448	0.447	0.440	0.469	0.455	0.456	0.452
ETTh2	96	**0.275**	**0.327**	0.281	0.332	0.290	0.338	0.289	0.342	0.295	0.350	0.297	0.349	0.289	0.341	0.302	0.348	0.333	0.387
	192	**0.341**	**0.375**	0.367	0.385	0.379	0.393	0.382	0.400	0.381	0.396	0.380	0.400	0.372	0.392	0.388	0.400	0.477	0.476
	336	0.392	0.412	0.394	0.418	**0.377**	**0.407**	0.424	0.439	0.426	0.438	0.428	0.432	0.386	0.414	0.426	0.433	0.594	0.541
	720	0.404	0.429	**0.395**	**0.420**	0.412	0.433	0.433	0.457	0.431	0.446	0.427	0.445	0.412	0.434	0.431	0.446	0.831	0.657
	Avg.	**0.353**	**0.386**	0.359	0.389	0.364	0.393	0.382	0.410	0.383	0.408	0.383	0.406	0.365	0.395	0.387	0.407	0.559	0.515
Weather	96	**0.146**	**0.187**	0.156	0.204	0.181	0.221	0.185	0.241	0.166	0.213	0.174	0.214	0.163	0.209	0.177	0.218	0.196	0.255
	192	**0.200**	**0.239**	0.203	0.245	0.228	0.260	0.237	0.287	0.213	0.254	0.221	0.254	0.208	0.250	0.225	0.259	0.237	0.296
	336	0.261	**0.284**	**0.250**	**0.284**	0.280	0.297	0.290	0.327	0.269	0.294	0.278	0.296	0.251	0.287	0.278	0.297	0.283	0.335
	720	**0.322**	0.352	0.338	**0.337**	0.355	0.345	0.367	0.380	0.346	0.343	0.358	0.347	0.339	0.341	0.354	0.348	0.345	0.381
	Avg.	**0.232**	**0.266**	0.237	0.268	0.261	0.281	0.270	0.309	0.249	0.276	0.258	0.278	0.240	0.272	0.259	0.280	0.265	0.317
Electricity	96	**0.140**	**0.238**	0.141	**0.238**	0.175	0.269	0.152	0.257	0.200	0.278	0.148	0.240	0.153	0.247	0.181	0.270	0.197	0.282
	192	**0.153**	0.253	0.157	**0.251**	0.184	0.273	0.156	0.257	0.200	0.280	0.162	0.253	0.166	0.256	0.188	0.274	0.196	0.285
	336	0.174	0.272	0.174	**0.268**	0.199	0.288	**0.171**	0.272	0.214	0.295	0.178	0.269	0.185	0.277	0.204	0.293	0.209	0.301
	720	0.213	0.306	**0.209**	0.303	0.235	0.319	0.221	**0.300**	0.255	0.327	0.225	0.317	0.225	0.310	0.246	0.324	0.245	0.333
	Avg.	**0.170**	0.267	**0.170**	**0.265**	0.198	0.287	0.175	0.272	0.217	0.295	0.178	0.270	0.182	0.273	0.205	0.290	0.212	0.300
Traffic	96	0.435	0.287	0.420	0.288	0.521	0.356	**0.388**	**0.265**	0.651	0.391	0.395	0.268	0.462	0.285	0.462	0.295	0.650	0.396
	192	0.452	0.298	0.434	0.298	0.505	0.341	**0.400**	0.278	0.602	0.363	0.417	**0.276**	0.473	0.296	0.466	0.296	0.598	0.370
	336	0.473	0.314	0.468	0.301	0.528	0.355	**0.411**	0.287	0.609	0.366	0.433	**0.283**	0.498	0.296	0.482	0.304	0.605	0.373
	720	0.505	0.336	0.487	0.309	0.566	0.373	**0.456**	0.305	0.647	0.385	0.467	**0.302**	0.506	0.313	0.514	0.322	0.645	0.394
	Avg.	0.466	0.309	0.452	0.299	0.530	0.356	**0.414**	0.284	0.627	0.376	0.428	**0.282**	0.485	0.297	0.481	0.304	0.625	0.383