CTIA-ODE: CONTINUOUS-TIME INTER-VARIABLE ATTENTION FOR LONG-TERM TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Although multivariate time series forecasting has broad application prospects, effectively incorporating exogenous variables remains challenging. Prevailing discrete-time architectures often impose rigid temporal alignment across heterogeneous variables, which may limit their ability to capture continuously evolving external effects. To address this limitation, we introduce CTIA-ODE, a continuous-time forecasting framework centered on Continuous-Time Inter-Variable Attention (CTIA). Specifically, CTIA is embedded within an ordinary differential equation (ODE) solver, enabling the continuously evolving endogenous trajectory to dynamically query a static exogenous context along the integration path without requiring explicit temporal alignment. To improve the stability of the continuous latent dynamics, we further introduce a dual-stabilization strategy: architecturally, a Linear Residual Augmentation (LRA) module provides a macroscopic trend anchor, while an Outlier-Robust Hybrid Loss (ORHL) mitigates the influence of heavy-tailed anomalies during optimization. Extensive experiments across eight real-world datasets demonstrate that CTIA-ODE achieves state-of-the-art forecasting performance across diverse datasets and prediction horizons, while maintaining a favorable accuracy-efficiency trade-off with a compact memory footprint and low inference latency, particularly under long-horizon forecasting settings. Anonymous code is available at `https://anonymous.4open.science/r/CTIA-ODE-15A6/`.

1 INTRODUCTION

Time series forecasting plays a crucial role in numerous key domains, including economics (Zhao et al., 2026), weather evolution (Pan et al., 2026) and healthcare management (Che et al., 2025). In real-world complex systems, such as power grids or financial markets, the severe fluctuations of target endogenous variables are fundamentally driven by external factors like holidays, macroeconomic indicators (Carriero et al., 2024), or extreme weather (Zhao et al., 2026; Zhou et al., 2025; Dong et al., 2026). Ignoring these exogenous variables in forecasting models directly caps the predictive upper bound, as they cannot anticipate systemic state shifts brought by external shocks (Wang et al., 2024; Alaali, 2026).

The mainstream approaches in recent years have mainly been based on Recurrent Neural Networks (RNNs) (Lin et al., 2025) and Transformers (Zhao et al., 2026; Liu et al., 2025; Chen et al., 2025b). RNNs rely on the step-by-step update of hidden states to process sequences, smoothly capturing dynamic temporal dependencies (Alharthi & Mahmood, 2024; Sun et al., 2026). Transformer architectures utilize global self-attention mechanisms to achieve parallel computation across time steps, demonstrating exceptional expressive power in capturing long-range temporal correlations (Zhao et al., 2026; Soe & Win, 2026). However, when attempting to simultaneously capture cross-correlations among multiple variables, both paradigms frequently fall into over-parameterization (Li & Lin, 2024). Real-world data is often limited in samples and accompanied by unpredictable noise(Godahewa et al., 2021); overly complex dependency modeling mechanisms and excessively deep network architectures easily lead to severe overfitting (Zhao et al., 2026; Zhou et al., 2025; Huang et al., 2023). This not only degrades the model's generalization ability but also causes precious high-frequency dynamic information to be over-smoothed (Hu et al., 2026). This issue is

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.

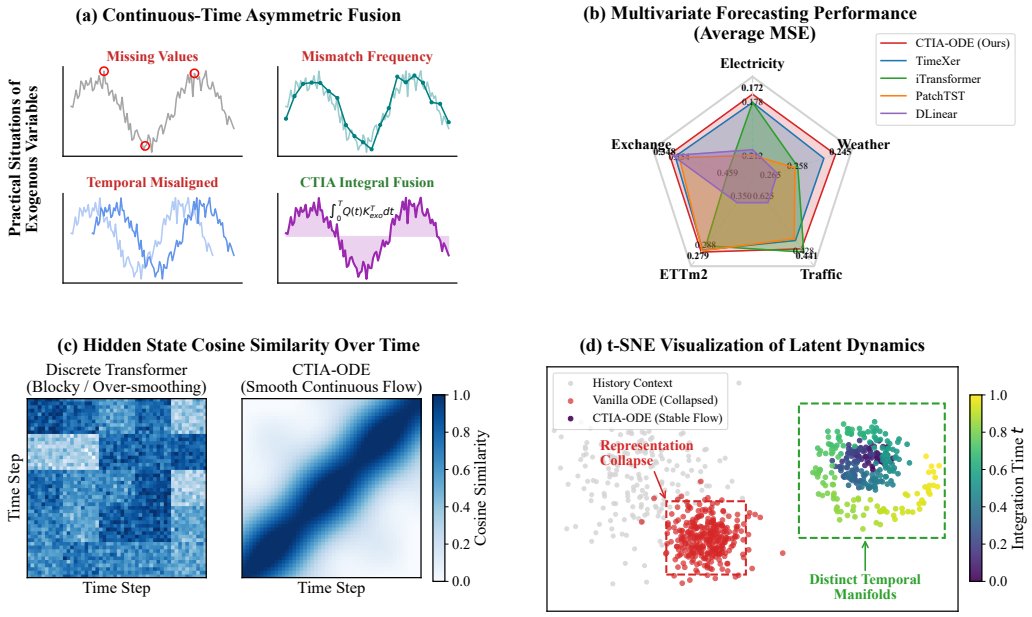

Figure 1: Motivation and empirical characteristics of CTIA-ODE. (a) Real-world exogenous variables may exhibit missing values, temporal misalignment, and frequency mismatch, motivating an asymmetric continuous-time fusion paradigm. (b) Forecasting performance on representative benchmarks. (c) Evolution of hidden-state similarity over the integration process, illustrating the difference between discrete representations and the continuous latent dynamics of CTIA-ODE. (d) t-SNE visualization of the learned latent states, providing a qualitative view of their temporal organization and representation stability.

not limited to prediction accuracy, but also concerns how latent representations evolve during forecasting. As illustrated in Figure 1(c–d), we therefore examine both the temporal evolution and latent-space organization of hidden states, providing complementary evidence for the continuous dynamics modeled by CTIA-ODE.

Beyond latent representation dynamics, incorporating exogenous variables introduces another layer of difficulty. Exogenous and endogenous variables possess distinctly different physical meanings (Zhou et al., 2025; Bi et al., 2024). Conflating them indiscriminately not only creates massive computational redundancy but also induces unnecessary reverse causal interference (Cheong et al., 2026; Zhou et al., 2025; Wang et al., 2024). Therefore, striking a balance between endogenous and exogenous features is particularly important. In past deep learning research, the many-to-one forecasting paradigm with exogenous variables was insufficiently explored(Han et al., 2026; Zhou et al., 2025), but recently this core pain point has rapidly become a major focus in the time series community (Zhou et al., 2025). Cutting-edge methods have proposed various fusion strategies. TimeXer employs an asymmetric representation granularity, utilizing global endogenous tokens to interact with exogenous variable tokens (Wang et al., 2024). CrossLinear introduces a cross-correlation embedding module to integrate external information (Zhou et al., 2025). ChronosX incorporates covariates into pretrained time series models (Cheong et al., 2026).

Despite significant progress, existing methods still face important limitations in incorporating exogenous information. To circumvent variable frequency mismatches, many approaches reduce the role of exogenous variables to simple auxiliary contextual representations (Liu et al., 2026; Wu et al., 2025; Zhou et al., 2025). However, a more fundamental challenge arises from the discrete architectures commonly used for multivariate forecasting (Bartolucci et al., 2019; Zhang et al., 2024). When heterogeneous endogenous and exogenous sequences are forced into strict step-by-step temporal alignment, substantial computational redundancy arises, while the resulting symmetric interactions frequently introduce unnecessary interference between primary and auxiliary variables (Yuan et al., 2026; Habiba et al., 2026). Consequently, conventional discrete attention mechanisms have difficulty modeling continuous interactions between an evolving endogenous trajectory and heterogeneous exogenous information without explicit temporal alignment (Wang et al., 2024).

Based on the above analysis, this study introduces a continuous-time perspective based on ordinary differential equations (ODEs) (Rubanova et al., 2019; Kidger et al., 2020) and proposes CTIA-ODE, a continuous-time forecasting architecture centered on Continuous-Time Inter-Variable Attention (CTIA). The model evolves the latent endogenous representation through an ODE solver while allowing the continuously evolving trajectory to dynamically generate query representations along the integration path. These queries interact with a static exogenous context constructed from external variables, enabling continuous-time cross-variable interaction without explicit step-by-step temporal alignment. To enhance the stability of the resulting latent dynamics, CTIA-ODE further incorporates a Linear Residual Augmentation (LRA) module and an Outlier-Robust Hybrid Loss (ORHL). As illustrated in Figure 1, this asymmetric design separates the evolving endogenous trajectory from the exogenous context while allowing information to flow continuously from the latter to the former.

Our primary contributions are summarized as follows:

- We propose CTIA-ODE, a continuous-time forecasting architecture centered on Continuous-Time Inter-Variable Attention (CTIA). This asymmetric mechanism enables the continuously evolving endogenous trajectory to dynamically query a static exogenous context, achieving cross-variable interaction without explicit temporal alignment or quadratic temporal complexity.
- We introduce a dual-stabilization strategy to safeguard the continuous latent dynamics. Architecturally, a Linear Residual Augmentation (LRA) module provides a macroscopic trend anchor; during optimization, an Outlier-Robust Hybrid Loss (ORHL) suppresses gradient spikes and mitigates the influence of heavy-tailed anomalies, significantly improving the robustness of the learned vector field.
- Comprehensive evaluations across eight real-world benchmarks demonstrate that CTIA-ODE achieves state-of-the-art performance. Beyond superior predictive accuracy, our framework exhibits a favorable accuracy-efficiency trade-off, operating with a compact memory footprint and low inference latency, particularly under extreme long-horizon settings.

2 Related Work

2.1 Continuous-Time Forecasting Models

Recent advancements in multivariate time series forecasting have been largely driven by discrete-time deep architectures, including efficient Transformers (Zhou et al., 2021; Wu et al., 2021; Nie et al., 2023; Zhang et al., 2026) and linear decomposition frameworks (Zeng et al., 2023; Li et al., 2023). While linear models efficiently extract macroscopic trends, these discrete paradigms lack continuous physical priors, making them susceptible to error compounding over extended horizons. Neural ODEs (Chen et al., 2018) address this by formulating hidden state transitions as continuous vector fields(Hasani et al., 2021; Gu et al., 2021). However, applying classical Neural ODEs or recent state-space variants (Chen et al., 2026; Dupont et al., 2019) directly to multivariate forecasting often leads to representation collapse, as a single solver struggles to handle both global trends and high-frequency spatial dynamics. Unlike these approaches(Massaroli et al., 2020; Dupont et al., 2019), CTIA-ODE explicitly decouples the learning process. By employing the LRA module to anchor macroscopic trends and utilizing zero-initialization for the continuous branch, our architecture maintains stable vector fields and prevents error amplification over long trajectories.

2.2 Robust Multivariate Dependencies

Balancing inter-series correlations with distributional robustness remains a primary challenge in multivariate forecasting(Chen et al., 2025a; Kim et al., 2025). Channel-dependent architectures (Zhang & Yan, 2023; Liu et al., 2024) capture cross-variate interactions but are prone to negative transfer when integrating heterogeneous variables, a vulnerability that asymmetric paradigms (Wang et al., 2024) aim to mitigate. Additionally, real-world time series frequently exhibit heavy-tailed distribution shifts and non-stationarity (Hu et al., 2026; Fan et al., 2023; Liu et al., 2022). While instance normalization (Kim et al., 2021) alleviates basic mean-variance drifts, standard quadratic loss functions over-penalize outliers(Shang et al., 2025; Gao et al., 2023), severely distorting the

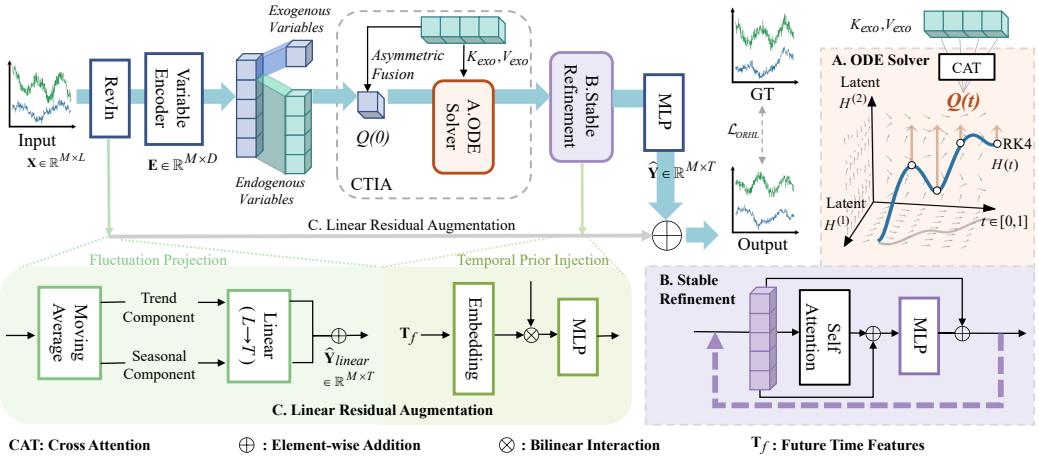

Figure 2: The overall framework of the proposed CTIA-ODE. (A) ODE Solver drives the continuous-time evolution of hidden states via RK4 integration, dynamically querying the static exogenous field through CTIA. (B) Stable Refinement performs discrete filtering on terminal states to maintain manifold stability. (C) Linear Residual Augmentation provides a robust prediction baseline by decoupling historical fluctuations and injecting future time priors.

continuous vector fields in dynamical models(Oreshkin et al., 2019; Le Guen & Thome, 2019). To address these interconnected challenges, CTIA-ODE introduces an asymmetric CTIA mechanism that selectively absorbs exogenous contexts without inducing symmetric cross-talk. Furthermore, we employ the Outlier-Robust Hybrid Loss, which combines localized quadratic convergence with a bounded linear penalty for large errors. This formulation provides strictly bounded gradients, enhancing optimization robustness against heavy-tailed sensor anomalies and helping to mitigate solver instability.

3 Methodology

3.1 Problem Formulation

Given a historical observation sequence $\mathcal{X} \in \mathbb{R}^{M \times L}$, the multivariate forecasting objective is to predict future values for all variables $\mathcal{Y} \in \mathbb{R}^{M \times T}$ over a horizon T. To model asymmetric cross-variable interactions, we designate one feature as the primary variable and use the remaining variables as auxiliary contextual variables. The remaining $M-1$ variables are encoded as a shared auxiliary context $X_{ex} \in \mathbb{R}^{K_{ex} \times L}$, where $K_{ex} = M - 1$. The model learns a mapping function $\mathcal{F}_\theta$:

$$\hat{\mathcal{Y}} = \mathcal{F}_\theta(\mathcal{X}, X_{ex}, T_f), \tag{1}$$

where T_f represents the known future time features and θ denotes learnable parameters. During integration, all M endogenous trajectories dynamically query this shared context, yielding the final multivariate prediction.

3.2 Structure Overview

As illustrated in Figure 2, the main path of the proposed CTIA-ODE architecture is based on the continuous-time integration method. The raw input sequence $X \in \mathbb{R}^{M \times L}$ is first normalized via the RevIN module (Kim et al., 2021). Subsequently, a simple variable encoder maps the sequence into a high-dimensional latent space, yielding representations $E \in \mathbb{R}^{M \times D}$. These dense representations are then decoupled into endogenous variables and exogenous variables.

In the core prediction workflow, the decoupled representations enter the CTIA module. An initial Asymmetric Fusion operates on these variables. Subsequently, the ODE Solver models the continuous-time evolution of the hidden state $H(t)$ over the integration interval $t \in [0, 1]$. The terminal state derived from the ODE solver is then fed into a Stable Refinement module. This block

stabilizes the latent manifold and filters integration noise before passing the representations through the final Multi-Layer Perceptron (MLP). Operating in parallel, the LRA module provides prediction enhancement by performing two different linear interaction mechanisms on distinct inputs. Finally, the outputs from both pathways are aggregated via addition and subsequently denormalized to produce the ultimate prediction $\hat{Y}$. The entire framework is optimized end-to-end guided by the Outlier-Robust Hybrid Loss ($\mathcal{L}_{ORHL}$).

3.3 Continuous-Time Integral Attention

We formulate the temporal evolution of the endogenous variables as a continuous-time dynamical system. To seamlessly incorporate the forcing exerted by exogenous variables, we propose CTIA. Unlike standard discrete attention applied at isolated network layers, CTIA evaluates cross-variable dependencies continuously along the integration trajectory.

Let E_{en} and E_{ex} denote the encoded latent representations of the endogenous and exogenous variables, respectively. Prior to the ODE integration, we project E_{ex} into a time-invariant exogenous field, obtaining the static key and value representations:

$$K_{exo} = E_{ex}W_K, \quad V_{exo} = E_{ex}W_V, \tag{2}$$

where $W_K, W_V \in \mathbb{R}^{D\times d_k}$ are learnable projection weight matrices. Before entering the continuous solver, an initial Asymmetric Fusion is applied to construct the starting state of the endogenous system $H(0)$ by fusing E_{en} with the static exogenous field.

During the continuous ODE integration spanning $t \in [0, 1]$, the endogenous hidden state $H(t)$ acts as a moving agent. At any instantaneous time step t, it generates a dynamic query $Q(t) = H(t)W_Q$. The instantaneous exogenous forcing $\mathcal{C}(t)$ is thus computed via a continuous cross-attention mechanism:

$$\mathcal{C}(t) = \text{Softmax}\left(\frac{Q(t)K_{exo}^{\top}}{\sqrt{d_k}}\right)V_{exo}. \tag{3}$$

Ultimately, the derivative governing the evolution of the endogenous state is defined by superimposing the base dynamics with the dynamic exogenous forcing:

$$\frac{dH(t)}{dt} = f_\phi(H(t)) + \alpha\mathcal{C}(t), \tag{4}$$

where $f_\phi(\cdot)$ is a MLP parameterizing the base vector field, and α is a learnable scaling parameter. In this formulation, K_{exo} and V_{exo} provide a stable global context, while $Q(t)$ continuously queries this field to adapt the evolutionary trajectory of $H(t)$.

3.4 Stable Refinement

Following the ODE integration, the terminal latent state $H(1)$ inevitably accumulates numerical noise. To filter this noise while preserving critical temporal details, we introduce the weight-sharing Dual-Loop Stable Refinement module.

As illustrated in Figure 2(B), taking $H(1)$ as the initial input, the refinement process iteratively applies a Self-Attention layer and a MLP, both equipped with residual connections and layer normalization. We utilize a weight-sharing recurrent design constrained to exactly two iterations. This dual-loop mechanism acts as a latent double-check while keeping the parameter footprint to a minimum.

3.5 Linear Residual Augmentation

This auxiliary module consists of two specific mechanisms. First, the Fluctuation Projection takes the normalized historical sequence $\mathbf{X}_{norm}$ as input. It decouples the raw sequence into trend and seasonal components via a moving average operation. These two components are then independently mapped from the look-back window L to the forecasting horizon T through linear layers and aggregated.

Second, the Temporal Prior Injection takes both the known future time features $\mathbf{T}_f$ and the refined terminal state $\tilde{H}_{sr}^{(2)}$ (from the Stable Refinement block) as inputs. It encodes $\mathbf{T}_f$ and fuses it with

the refined latent manifold via a bilinear interaction mechanism to yield auxiliary temporal effects. Finally, the outputs from these two parallel mechanisms are superimposed onto the primary nonlinear prediction via learnable scalar weights to form the ultimate forecast.

3.6 Outlier-Robust Hybrid Loss

Real-world physical sensor data often contains non-stationary distributions and outliers. Training continuous dynamical systems under standard quadratic objectives can heavily penalize these anomalies, inducing large gradients that distort the learned vector field. To improve robustness, we optimize the aggregated final prediction $\hat{Y}$ using the Outlier-Robust Hybrid Loss ($\mathcal{L}_{\text{ORHL}}$). Given the ground truth Y, let $e = \hat{Y} - Y$ denote the forecasting residual. The ORHL applies a piecewise penalty function element-wise:

$$\mathcal{L}_{\text{ORHL}}(e) = \frac{1}{N}\sum_{i=1}^{N} \begin{cases} \frac{1}{2}e_i^2 + \lambda|e_i|, & \text{if } |e_i| \le \delta, \\ (\delta+\lambda)|e_i| - \frac{1}{2}\delta^2, & \text{otherwise,} \end{cases} \tag{5}$$

where δ dictates the transition threshold and λ acts as a linear scaling factor.

This formulation provides two critical properties for continuous-time modeling. In the localized convergence zone ($|e_i| \le \delta$), the objective operates as an Elastic-Net-style composite: the quadratic curvature provides stable optimization, while the scaled ℓ_1 penalty encourages sparsity in the residual distribution. For large anomalous deviations ($|e_i| > \delta$), the gradient strictly saturates at a bounded constant $\pm(\delta + \lambda)$. This bounded prediction gradient limits the magnitude of loss signals entering the continuous branch and improves optimization robustness under large residuals.

4 Experiments

We evaluate CTIA-ODE on multiple real-world multivariate time series forecasting benchmarks. The experiments are designed to answer the following research questions: **RQ1:** How does CTIA-ODE compare against state-of-the-art (SOTA) forecasters across varying prediction horizons? **RQ2:** What are the individual contributions of the Outlier-Robust Hybrid Loss (ORHL), the LRA module, and the continuous-time attention mechanism? **RQ3:** Does the model achieve a favorable trade-off between computational efficiency and forecasting robustness?

4.1 Experimental Settings

Datasets. We conduct experiments on eight widely used real-world multivariate benchmarks: Electricity, Exchange, Traffic, Weather, and the ETT datasets (ETTh1, ETTh2, ETTm1, ETTm2). These datasets encompass diverse domains, ranging from energy management to meteorological forecasting, with varying periodicities and dimensionalities.

Baselines and Evaluation Metrics. We comprehensively evaluate CTIA-ODE against nine state-of-the-art multivariate forecasting models, which can be categorized into four distinct architectural paradigms: (1) Transformer-based models: TimeXer (Wang et al., 2024), iTransformer (Liu et al., 2024), PatchTST (Nie et al., 2023), Crossformer (Zhang & Yan, 2023), and Autoformer (Wu et al., 2021); (2) State-Space Models (SSMs): S-Mamba(Wang et al., 2025); (3) Linear models: DLinear (Zeng et al., 2023); (4) CNN and Time-Frequency architectures: TimesNet (Wu et al., 2023) and WaveTS-B(Zhou et al., 2026). To quantitatively evaluate the forecasting performance, we use the standard Mean Squared Error (MSE) and Mean Absolute Error (MAE).

Implementation Details. Following the standard protocol in Multivariate Time Series Forecasting, the look-back window length L is fixed to 96, and the prediction horizon T is evaluated at $\{96, 192, 336, 720\}$. We utilize the AdamW optimizer with an initial learning rate of 5×10^{-4} and a weight decay of 1×10^{-4}. The learning rate is dynamically adjusted using a ReduceLROnPlateau scheduler with a decay factor of 0.5 and a patience of 3 epochs, bounded by a minimum learning rate of 1×10^{-6}. To strike an optimal balance between representational capacity and parameter efficiency, the hidden dimension d_model is set to 128, completely avoiding the heavy memory overhead of typical 512-dimensional Transformers. The network is trained using our proposed ORHL with an early stopping patience of 12 epochs.

Table 1: Multivariate forecasting results. The look-back length L is set to 96. The best results are highlighted in **bold** and the second best are underlined.

Models		**CTIA-ODE (Ours)**		WaveTS-B		S-Mamba		TimeXer		iTransformer		PatchTST		DLinear		Crossformer		TimesNet		Autoformer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Electricity	96	**0.140**	**0.233**	0.143	0.235	0.147	0.249	0.147	0.251	0.148	0.240	0.195	0.285	0.197	0.282	0.219	0.314	0.168	0.272	0.201	0.317
	192	**0.160**	**0.252**	0.164	0.255	0.167	0.270	0.163	0.266	0.162	0.253	0.199	0.289	0.196	0.285	0.231	0.322	0.184	0.289	0.222	0.334
	336	0.181	0.274	0.185	0.278	0.189	0.291	0.184	0.282	**0.178**	**0.269**	0.215	0.305	0.209	0.301	0.246	0.337	0.198	0.300	0.231	0.338
	720	**0.205**	**0.296**	0.211	0.302	0.232	0.327	0.220	0.312	0.225	0.317	0.256	0.337	0.245	0.333	0.280	0.363	0.220	0.320	0.254	0.361
	Avg	**0.172**	**0.264**	0.176	0.268	0.183	0.284	0.178	0.278	0.178	0.270	0.216	0.304	0.212	0.300	0.244	0.334	0.192	0.295	0.227	0.338
Exchange	96	0.093	0.215	0.098	0.219	0.089	0.211	0.101	0.226	0.118	0.253	**0.088**	**0.205**	**0.088**	0.218	0.256	0.367	0.107	0.234	0.197	0.323
	192	0.188	0.308	0.192	0.311	0.187	0.309	0.199	0.317	0.232	0.355	**0.176**	**0.299**	**0.176**	0.315	0.470	0.509	0.226	0.344	0.300	0.369
	336	0.325	0.415	0.342	0.428	0.338	0.423	0.334	0.424	0.434	0.489	**0.301**	**0.397**	0.313	0.427	1.268	0.883	0.367	0.448	0.509	0.524
	720	**0.788**	**0.668**	1.031	0.768	0.875	0.710	0.803	0.680	1.050	0.750	0.901	0.714	0.839	0.695	1.767	1.068	0.964	0.746	1.447	0.941
	Avg	**0.348**	**0.402**	0.415	0.431	0.372	0.413	0.359	0.411	0.459	0.463	0.367	0.404	0.354	0.414	0.940	0.707	0.416	0.443	0.613	0.539
Traffic	96	0.411	**0.253**	0.416	0.271	0.428	0.271	0.435	0.279	**0.395**	0.268	0.462	0.295	0.650	0.396	0.522	0.290	0.593	0.321	0.613	0.388
	192	0.425	**0.262**	0.431	0.278	0.442	0.280	0.457	0.288	**0.417**	0.276	0.466	0.296	0.598	0.370	0.530	0.293	0.617	0.336	0.616	0.382
	336	0.456	**0.270**	0.462	0.281	0.471	0.292	0.483	0.296	**0.433**	0.283	0.482	0.304	0.605	0.373	0.558	0.305	0.629	0.336	0.622	0.337
	720	0.474	**0.285**	0.481	0.302	0.495	0.308	0.522	0.316	**0.467**	0.302	0.514	0.322	0.645	0.394	0.589	0.328	0.640	0.350	0.660	0.408
	Avg	0.441	**0.268**	0.448	0.283	0.459	0.288	0.474	0.293	**0.428**	0.282	0.481	0.304	0.625	0.383	0.550	0.304	0.620	0.336	0.628	0.379
Weather	96	**0.153**	**0.191**	0.172	0.225	0.169	0.210	0.165	0.214	0.174	0.214	0.177	0.218	0.196	0.255	0.158	0.230	0.172	0.220	0.266	0.336
	192	0.208	**0.247**	0.215	0.261	0.214	0.253	0.211	0.253	0.221	0.254	0.225	0.259	0.237	0.296	**0.206**	0.277	0.219	0.261	0.307	0.367
	336	**0.267**	**0.288**	0.273	0.298	0.274	0.296	0.271	0.298	0.278	0.296	0.278	0.297	0.283	0.335	0.272	0.335	0.280	0.306	0.359	0.395
	720	0.351	**0.342**	0.358	0.352	0.353	0.348	0.346	0.348	0.358	0.349	0.354	0.348	**0.345**	0.381	0.398	0.418	0.365	0.359	0.419	0.428
	Avg	**0.245**	**0.267**	0.254	0.284	0.253	0.277	0.249	0.280	0.258	0.279	0.259	0.281	0.265	0.317	0.259	0.315	0.259	0.287	0.338	0.382
ETTh1	96	0.390	**0.397**	0.396	0.406	0.386	0.405	0.391	0.409	0.386	0.405	0.414	0.419	0.386	0.400	0.423	0.448	**0.384**	0.402	0.449	0.459
	192	0.455	0.437	0.462	0.443	0.443	0.437	**0.436**	0.445	0.441	0.436	0.460	0.445	0.437	0.432	0.471	0.474	**0.436**	**0.429**	0.500	0.482
	336	0.503	0.464	0.511	0.472	0.489	0.468	**0.476**	**0.455**	0.487	0.458	0.501	0.466	0.481	0.459	0.570	0.546	0.491	0.469	0.521	0.496
	720	0.526	0.504	0.534	0.512	0.502	0.489	**0.475**	**0.470**	0.503	0.491	0.500	0.488	0.519	0.516	0.653	0.621	0.521	0.500	0.514	0.512
	Avg	0.469	0.450	0.476	0.458	0.455	0.450	**0.445**	**0.444**	0.454	0.447	0.469	0.454	0.456	0.452	0.529	0.522	0.458	0.450	0.496	0.487
ETTh2	96	**0.290**	**0.338**	0.295	0.344	0.296	0.348	0.293	0.347	0.297	0.349	0.302	0.348	0.333	0.387	0.745	0.584	0.340	0.374	0.346	0.388
	192	0.383	0.396	0.389	0.403	0.376	0.396	**0.373**	**0.395**	0.380	0.400	0.388	0.400	0.477	0.476	0.877	0.656	0.402	0.414	0.456	0.452
	336	0.428	0.432	0.435	0.441	0.424	0.431	**0.422**	**0.430**	0.428	0.432	0.426	0.433	0.594	0.541	1.043	0.731	0.452	0.452	0.482	0.486
	720	0.433	0.444	0.441	0.452	0.426	0.444	**0.414**	**0.442**	0.427	0.445	0.431	0.446	0.831	0.657	1.104	0.763	0.462	0.468	0.515	0.511
	Avg	0.383	**0.402**	0.390	0.410	0.381	0.405	**0.375**	0.404	0.383	0.407	0.387	0.407	0.559	0.515	0.942	0.684	0.414	0.427	0.450	0.459
ETTm1	96	**0.326**	**0.351**	0.333	0.358	0.333	0.368	0.327	0.363	0.334	0.368	0.329	0.367	0.345	0.372	0.404	0.426	0.338	0.375	0.505	0.475
	192	0.378	**0.377**	0.385	0.384	0.376	0.390	0.370	0.389	0.377	0.391	**0.367**	0.385	0.380	0.389	0.450	0.451	0.374	0.387	0.553	0.496
	336	0.408	**0.399**	0.415	0.406	0.408	0.413	0.405	0.415	0.426	0.420	**0.399**	0.410	0.413	0.413	0.532	0.515	0.410	0.411	0.621	0.537
	720	0.472	**0.434**	0.481	0.442	0.475	0.448	0.459	0.450	0.491	0.459	**0.454**	0.439	0.474	0.453	0.666	0.589	0.478	0.450	0.671	0.561
	Avg	0.396	**0.390**	0.403	0.397	0.398	0.405	0.390	0.405	0.407	0.410	**0.387**	0.400	0.403	0.407	0.513	0.496	0.400	0.406	0.588	0.517
ETTm2	96	**0.175**	**0.253**	0.181	0.260	0.179	0.263	0.177	0.266	0.180	0.264	**0.175**	0.259	0.193	0.292	0.287	0.366	0.187	0.267	0.255	0.339
	192	**0.241**	**0.297**	0.248	0.305	0.250	0.309	0.245	0.306	0.250	0.309	**0.241**	0.302	0.284	0.362	0.414	0.492	0.249	0.309	0.281	0.340
	336	**0.300**	**0.334**	0.308	0.342	0.312	0.349	0.305	0.344	0.311	0.348	0.305	0.343	0.369	0.427	0.597	0.542	0.321	0.351	0.339	0.372
	720	0.401	**0.394**	0.411	0.402	0.411	0.406	**0.399**	0.402	0.412	0.407	0.402	0.400	0.554	0.522	1.730	1.042	0.408	0.403	0.433	0.432
	Avg	**0.279**	**0.319**	0.287	0.327	0.288	0.332	0.282	0.331	0.288	0.332	0.281	0.326	0.350	0.401	0.757	0.610	0.291	0.333	0.327	0.371
1st Count		**44**		0		0		15		7		12		3		1		3		0	
Avg 1st Count		**11**		0		0		3		1		1		0		0		0		0	

4.2 Main Results (RQ1)

Table 1 summarizes the multivariate forecasting performance across 8 representative benchmarks. A lower MSE or MAE indicates better predictive accuracy.

As shown in the table, our CTIA-ODE achieves the best performance on 44 out of the evaluated metrics, consistently outperforming the leading baselines TimeXer (15 out of 80) and PatchTST (12 out of 80). A key advantage of CTIA-ODE is its robustness under extreme long-term horizons (e.g., $T = 720$), where discrete-time models typically suffer from severe error accumulation. For example, on the ETTm1 dataset, CTIA-ODE maintains an MSE of 0.472 at $T = 720$ compared to 0.326 at $T = 96$, demonstrating a highly constrained error growth. This long-term stability across benchmarks like Electricity and ETTm2 validates our Augmented Continuous-Time Architecture: the LRA provides a macroscopic trend baseline to mitigate representation collapse, while the Neural ODE branch focuses on resolving local continuous dynamics.

Furthermore, CTIA-ODE effectively mitigates the negative transfer issue commonly observed in highly heterogeneous datasets (e.g., Exchange, Weather, and Traffic). While fully channel-dependent models such as Crossformer often suffer from non-causal cross-talk in these scenarios, CTIA-ODE addresses this through Continuous-Time Inter-Variable Attention. By processing exogenous variables asymmetrically as continuous contextual signals rather than computing symmetrical dense cross-attention, the model incorporates valid external constraints while filtering heterogeneous noise. Consequently, CTIA-ODE achieves the lowest MAE scores across all prediction horizons on the high-dimensional Traffic dataset, confirming its pointwise predictive reliability under complex spatial conditions.

4.3 Ablation Study (RQ2)

To evaluate the individual contributions of the proposed CTIA-ODE components, we conduct a detailed ablation study focusing on the Augmented Continuous-Time Architecture, the Continuous-

Table 2: Ablation study on Electricity, ETTm1, Traffic, and Weather datasets. The complete CTIA-ODE is compared against variants ablating the loss function, architecture, and attention mechanism. The best results are highlighted in **bold**.

Models	Electricity				ETTm1				Traffic				Weather			
	$T = 96$		$T = 720$		$T = 96$		$T = 720$		$T = 96$		$T = 720$		$T = 96$		$T = 720$	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
CTIA-ODE (Full)	**0.140**	**0.233**	**0.205**	**0.296**	**0.326**	**0.351**	**0.472**	**0.434**	**0.411**	**0.253**	**0.474**	**0.285**	**0.153**	**0.191**	**0.351**	**0.342**
w/o Linear (Pure ODE)	0.145	0.238	0.238	0.318	0.332	0.358	0.548	0.485	0.415	0.258	0.535	0.315	0.159	0.201	0.395	0.380
w/o ODE (Pure Linear)	0.149	0.245	0.215	0.302	0.341	0.365	0.488	0.448	0.430	0.270	0.492	0.300	0.165	0.208	0.368	0.355
w/o CTIA (Pure CI)	0.144	0.238	0.212	0.301	0.336	0.360	0.486	0.445	0.428	0.265	0.505	0.304	0.163	0.205	0.370	0.362
w/o ORHL (Pure MSE)	0.143	0.241	0.211	0.308	0.338	0.368	0.491	0.456	0.418	0.263	0.488	0.296	0.160	0.207	0.364	0.358
w/o ORHL (Pure MAE)	0.146	0.235	0.215	0.299	0.342	0.355	0.495	0.440	0.422	0.256	0.491	0.288	0.164	0.199	0.369	0.350

Time Inter-Variable Attention, and the ORHL. We compare the full model against five ablated variants on the Electricity, ETTm1, Traffic, and Weather datasets.

Table 2 presents the quantitative results across short ($T = 96$) and extended ($T = 720$) forecasting horizons. To validate the superiority of ORHL, we replace it with standard pure MSE and pure MAE losses. Predictive performance degrades under both single-metric configurations: pure MSE over-penalizes heavy-tailed anomalies, which distorts the continuous vector field, while pure MAE leads to slower convergence and suboptimal global optima. The proposed hybrid objective effectively mitigates both issues.

Furthermore, to analyze the augmented architecture, we decouple the framework into a Pure ODE variant (removing the LRA) and a Pure Linear variant (removing the continuous dynamics). The Pure Linear variant struggles to capture high-frequency complex variations, yielding consistently higher errors. Conversely, while the Pure ODE variant performs adequately on short horizons, it suffers from severe representation collapse at $T = 720$, evidenced by a nearly 16% increase in MSE on the ETTm1 dataset. This demonstrates that the LRA module is crucial for maintaining macroscopic trend stability, while the ODE branch is necessary for resolving local microscopic dynamics. Finally, removing the exogenous variable interactions to enforce absolute channel independence results in noticeable performance drops, particularly on heterogeneous datasets like Traffic. This confirms that the asymmetric continuous cross-attention successfully captures multivariate correlations without inducing negative transfer.

4.4 Model Efficiency and In-Depth Mechanistic Analysis (RQ3)

Figure 3(a) illustrates the trade-off between predictive accuracy and computational resources. Conventional Transformer architectures often incur high inference latency and substantial VRAM usage due to complex attention operations. In contrast, CTIA-ODE lies on the Pareto front, achieving low long-term prediction errors (e.g., 0.205 at $T = 720$) with a compact memory footprint and an inference speed of 6.8 ms per sample.

Figure 3(b) evaluates the iteration count (K) in the Looped Stable Block. The error exhibits a U-shaped curve, minimizing at $K = 2$. A single iteration ($K = 1$) yields suboptimal accuracy due to insufficient multivariate fusion, while increasing iterations ($K \geq 3$) progressively degrades performance. On the Weather dataset at $T = 96$, the MSE increases from 0.154 ($K = 2$) to 0.161 ($K = 5$). This degradation stems from over-smoothing, where excessive interactions blur distinct variable representations. We therefore use $K = 2$ to balance accuracy and efficiency.

Figure 3(c) highlights the robustness of ORHL. Under heavy-tailed perturbations, models trained with standard MSE exhibit increasing error, as the quadratic penalty amplifies localized gradient spikes. In contrast, ORHL maintains stable forecasting accuracy across contamination levels by enforcing bounded gradients for large anomalies.

Figure 3(d) analyzes continuous vector field trajectories during numerical integration. While an unconstrained continuous ODE can exhibit gradient instability and oscillations, our architecture maintains a smoother vector flow. This suggests that LRA handles macroscopic trend drift, reducing the burden on the ODE dynamics.

Beyond predictive accuracy, CTIA-ODE addresses the high GPU memory footprint typical of Transformer-based architectures. By using a compact hidden dimensionality ($d_{model} = 128$), the

model reduces state-transition parameters and operates within 4GB VRAM without OOM errors, improving practical scalability.

To stabilize early-stage training, we zero-initialize the final projection layer of the ODE state network. This sets the autonomous component of the vector field near zero, providing a "soft-start" for the continuous branch while the CTIA forcing remains controlled by its learnable scaling factor. This allows LRA to establish the macroscopic trend before stronger nonlinear ODE dynamics emerge, resulting in smoother loss convergence and more stable long-horizon errors.

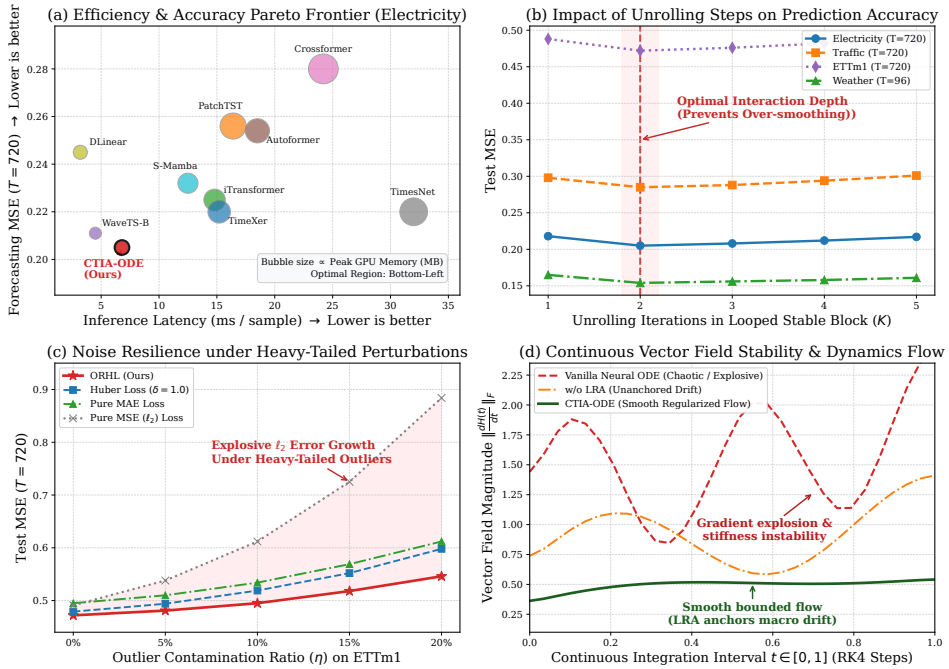

Figure 3: In-depth empirical and mechanistic analysis of CTIA-ODE. (a) Pareto frontier comparing inference latency, forecasting error ($T = 720$), and peak GPU memory footprint (represented by bubble size) on the Electricity benchmark. (b) Impact of unrolling iterations (K) on forecasting accuracy, highlighting the optimal interaction depth at $K = 2$ to prevent over-smoothing. (c) Resilience to heavy-tailed sensor anomalies under varying outlier contamination ratios η on the ETTm1 dataset. (d) Continuous vector field trajectory magnitude $\|\frac{dH(t)}{dt}\|_F$ along the RK4 integration interval $t \in [0, 1]$.

5 Conclusion

In this paper, we propose CTIA-ODE, an efficient and robust continuous-time framework for multivariate time series forecasting. To handle variable heterogeneity without forcing rigid step-by-step alignment, we design the asymmetric Continuous-Time Inter-Variable Attention (CTIA) mechanism. This paradigm enables the continuously evolving endogenous trajectory to dynamically query a static exogenous context, facilitating cross-variable interactions without explicit step-by-step temporal alignment and helping preserve informative representations over long horizons. To improve the stability of long-horizon forecasting under noisy observations, we introduce a dual-stabilization strategy: architecturally, a Linear Residual Augmentation (LRA) module anchors the long-term trend, while an Outlier-Robust Hybrid Loss (ORHL) bounds the prediction gradients and reduces the influence of heavy-tailed anomalies during optimization. Comprehensive evaluations across eight benchmarks demonstrate that CTIA-ODE achieves state-of-the-art performance across diverse datasets and forecasting horizons. It further provides a favorable accuracy-efficiency trade-off, with a compact memory footprint and low inference latency. Future work will explore the extension of the CTIA mechanism to irregularly sampled multivariate time series and event-driven forecasting scenarios.

AI USE STATEMENT

During the preparation of this manuscript, we utilized a Large Language Model (LLM) to assist with copy-editing and proofreading. Specifically, the LLM was employed exclusively to refine sentence structures, correct grammatical errors, and enhance the overall fluency and readability of the text. The authors carefully reviewed all AI-assisted revisions and assume full responsibility for the scientific content, experimental results, and final presentation of this work.

ETHICS STATEMENT

Since this research focuses fundamentally on time series forecasting methodology, it does not involve sensitive data and presents no potential ethical risks.

REPRODUCIBILITY STATEMENT

The mathematical formulation of our model is formally defined in the main text. All implementation details, encompassing dataset specifications, evaluation metrics, experimental configurations, and the complete source code, are provided in the supplementary material.

REFERENCES

Hasan Alaali. Beyond time series: Reconceptualizing time as a causal mechanism in organizational decision-making. *Available at SSRN 7256798*, 2026.

Musleh Alharthi and Ausif Mahmood. Enhanced linear and vision transformer-based architectures for time series forecasting. *Big Data and Cognitive Computing*, 8(5):48, 2024.

Lorenzo Bartolucci, Stefano Cordiner, Vincenzo Mulone, and Marina Santarelli. Hybrid renewable energy systems: Influence of short term forecasting on model predictive control performance. *Energy*, 172:997–1004, 2019.

Xin Bi, Qinghan Jin, Meiling Song, Xin Yao, Xiangguo Zhao, Ye Yuan, and Guoren Wang. Spatiotemporal learning with decoupled causal attention for multivariate time series. *IEEE Transactions on Big Data*, 11(4):1589–1599, 2024.

Andrea Carriero, Davide Pettenuzzo, and Shubhranshu Shekhar. Macroeconomic forecasting with large language models. *arXiv preprint arXiv:2407.00890*, 2024.

Jinshan Che, Mingming Sun, Yuhong Wang, and Zhendan Xu. Fusion-driven multimodal learning for biomedical time series in surgical care. *Frontiers in Physiology*, 16:1605406, 2025.

Junfeng Chen, Azhu Guan, Jingjing Du, and Altangerel Ayush. Multivariate time series prediction with multi-feature analysis. *Expert Systems with Applications*, 268:126302, 2025a.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Xingsheng Chen, Regina Zhang, Bo Gao, Xingwei He, Xiaofeng Liu, Pietro Lio, Kwok-Yan Lam, and Siu-Ming Yiu. Mode: Efficient time series prediction with mamba enhanced by low-rank neural odes. *arXiv preprint arXiv:2601.00920*, 2026.

Yu Chen, Nathalia Céspedes, and Payam Barnaghi. A closer look at transformers for time series forecasting: Understanding why they work and where they struggle. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 7763–7780, 2025b.

Wei Soon Cheong, Lian Lian Jiang, and Jamie Ng Suat Ling. Assessing electricity demand forecasting with exogenous data in time series foundation models. *arXiv preprint arXiv:2602.05390*, 2026.

Zongxin Dong, Shuangshuang Li, Guangyuan Pan, Zhenghong Bi, and Chuanxia Hu. Ftsformer: A financial-initiated multi-scale transformer with exogenous fusion for time series forecasting. *International Journal of Machine Learning and Cybernetics*, 17(9):443, 2026.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Wei Fan, Pengyang Wang, Dongkun Wang, Dongjie Wang, Yuanchun Zhou, and Yanjie Fu. Dish-ts: A general paradigm for alleviating distribution shift in time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7522–7529, 2023.

Yunlong Gao, Yuzhe Feng, Youwei Xie, Jinyan Pan, and Feiping Nie. Normalized robust pca with adaptive reconstruction error minimization. *IEEE Transactions on Knowledge and Data Engineering*, 36(6):2587–2599, 2023.

Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I. Webb, Rob J. Hyndman, and Pablo Montero-Manso. Monash time series forecasting archive. *arXiv preprint arXiv:2105.06643*, 2021.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021.

Mansura Habiba, Barak A. Pearlmutter, and Mehrdad Maleki. Emerging trends and open challenges in time series modeling. In *Recent Trends in Modelling the Continuous Time Series Using Deep Learning*, pp. 63–93. Springer, 2026.

Yongming Han, Qianni Zhang, Ying Wang, Xintian Wang, Yuhang Hao, Yi Cai, Xingxing Zhang, and Zhiqiang Geng. Causal-inspired regularization and dynamic embeddings network for time series forecasting and energy saving of household electricity. *Energy and Buildings*, pp. 117926, 2026.

Ramin Hasani, Mathias Lechner, Alexander Amini, Daniela Rus, and Radu Grosu. Liquid time-constant networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7657–7666, 2021.

Yifan Hu, Jie Yang, Tian Zhou, Peiyuan Liu, Yujin Tang, Rong Jin, and Liang Sun. Bridging past and future: Distribution-aware alignment for time series forecasting. In *International Conference on Learning Representations*, 2026.

Qihe Huang, Lei Shen, Ruixin Zhang, Shouhong Ding, Binwu Wang, Zhengyang Zhou, and Yang Wang. Crossgnn: Confronting noisy multivariate time series via cross interaction refinement. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46885–46902, 2023.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6696–6707, 2020.

Jongseon Kim, Hyungjoon Kim, HyunGi Kim, Dongjun Lee, and Sungroh Yoon. A comprehensive survey of deep learning for time series forecasting: Architectural diversity and open challenges. *Artificial Intelligence Review*, 58:216, 2025.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2021.

Vincent Le Guen and Nicolas Thome. Shape and time distortion loss for training deep time series forecasting models. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yicheng Li and Qian Lin. Improving adaptivity via over-parameterization in sequence models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52438–52480, 2024.

Zhe Li, Shiyi Qi, Yiduo Li, and Zenglin Xu. Revisiting long-term time series forecasting: An investigation on linear mapping. *arXiv preprint arXiv:2305.10721*, 2023.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Feiyu Zhao, Ruichao Mo, and Haotong Zhang. Segrnn: Segment recurrent neural network for long-term time series forecasting. *IEEE Internet of Things Journal*, 2025.

Kun Liu, Renjun Jia, Ruifeng Yang, Xirui Zeng, Yuqi Liang, and Cen Chen. Are global dependencies necessary? scalable time series forecasting via local cross-variate modeling. In *International Conference on Learning Representations*, 2026.

Qiang Liu, Shi Li, Genfu Zhou, Zhonglun Feng, Gang Wang, and Qingtai Qiu. Attribution of nonstationary changes in the annual runoff of the weihe river using the de-nonstationarity method. *Hydrology Research*, 53:407–418, 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 *International Conference on Learning Representations*, 2024.

Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer-xl: Long-context transformers for unified time series forecasting. In *International Conference on Learning Representations*, 2025.

Stefano Massaroli, Michael Poli, Jinkyoo Park, Atsushi Yamashita, and Hajime Asama. Dissecting neural odes. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3952–3963, 2020.

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.

Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019.

Xuanwen Pan, Yoichi Shimomura, Sichen Tao, Masatoschi Kawai, Keichi Takahashi, and Hiroyuki Takizawa. Cityscalecast: Spatiotemporal gnn for city-scale weather prediction with graphcast-guided parallel modeling and multi-step forecasting in sendai. In *Proceedings of the Supercomputing Asia and International Conference on High Performance Computing in Asia Pacific Region Workshops*, pp. 177–185, 2026.

Yulia Rubanova, Ricky T. Q. Chen, and David K. Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Wenlong Shang, Shihao Tian, Xutong Wan, and Peng Chang. Cognos: Universal enhancement for time series anomaly detection via constrained gaussian-noise optimization and smoothing. *arXiv preprint arXiv:2511.06894*, 2025.

Mar Mar Soe and Si Si Mar Win. Adaptive scaling-enhanced dual attention postlstm model for time series data. In *2026 IEEE Conference on Computer Applications*, pp. 1–6. IEEE, 2026.

Chenxi Sun, Moxian Song, Derun Cai, Baofeng Zhang, Hongyan Li, and Shenda Hong. A review of deep learning methods for irregularly sampled medical time series data. *Health Data Science*, 6:0456, 2026.

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, 2024.

Zihan Wang, Fanheng Kong, Shi Feng, Ming Wang, Xiaocui Yang, Han Zhao, Daling Wang, and Yifei Zhang. Is mamba effective for time series forecasting? *Neurocomputing*, 619:129178, 2025.

Binqing Wu, Zongjiang Shang, Jianlong Huang, and Ling Chen. Millgnn: Learning multi-scale lead-lag dependencies for multi-variate time series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 3344–3354, 2025.

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.

Wei Yuan, Chaoqun Yang, Xiangyu Zhao, Quoc Viet Hung Nguyen, Yang Cao, Tieke He, and Hongzhi Yin. Tackling data heterogeneity in federated time series forecasting. *Science China Information Sciences*, 69(5):152102, 2026.

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.

Ke Zhang, Yinghong Gao, and Meng Li. Asymmetric attention model for ride-sharing problem based on time-varying road networks. In *International Conference on Artificial Intelligence and Autonomous Transportation*, pp. 281–288. Springer, 2024.

Qianru Zhang, Honggang Wen, Ming Li, Dong Huang, Siu-Ming Yiu, Christian S. Jensen, and Pietro Liò. Autohformer: Efficient hierarchical autoregressive transformer for time series prediction. In *2026 IEEE 42nd International Conference on Data Engineering*, pp. 183–196. IEEE, 2026.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *International Conference on Learning Representations*, 2023.

Jingyuan Zhao, Fulin Chu, Lili Xie, Yunhong Che, Yuyan Wu, and Andrew F. Burke. A survey of transformer networks for time series forecasting. *Computer Science Review*, 60:100883, 2026.

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.

Pengfei Zhou, Yunlong Liu, Junli Liang, Qi Song, and Xiangyang Li. Crosslinear: Plug-and-play cross-correlation embedding for time series forecasting with exogenous variables. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025.

Zheng Zhou, Yu-Jie Xiong, Jia-Chen Zhang, and Chun-Ming Xia. Wavelet mixture of experts for time series forecasting. *Expert Systems with Applications*, 323:132399, 2026.

APPENDIX CONTENT

A DETAILED DATASET DESCRIPTIONS

To comprehensively evaluate the robustness and generalization capability of CTIA-ODE, we utilize eight real-world multivariate time series datasets spanning diverse physical and social domains. The statistical details of these benchmarks are summarized in Table 3.

Table 3: Summary of the eight real-world multivariate time series datasets used in our experiments.

Dataset	Domain	Variables (M)	Timesteps	Granularity	Prediction Task
Electricity	Energy	321	26,304	1 Hour	Power consumption forecasting
Traffic	Transportation	862	17,544	1 Hour	Road occupancy rate prediction
Weather	Meteorology	21	52,696	10 Minutes	Climatological indicator tracking
Exchange	Finance	8	7,588	1 Day	Currency exchange rate prediction
ETTh1	Energy (Transformer)	7	17,420	1 Hour	Oil temperature and load forecasting
ETTh2	Energy (Transformer)	7	17,420	1 Hour	Oil temperature and load forecasting
ETTm1	Energy (Transformer)	7	69,680	15 Minutes	Oil temperature and load forecasting
ETTm2	Energy (Transformer)	7	69,680	15 Minutes	Oil temperature and load forecasting

- **Electricity:** Captures the hourly electricity consumption of 321 clients from 2012 to 2014. It features strong daily and weekly periodicities with high spatial correlations among clients.
- **Traffic:** Contains hourly road occupancy rates measured by 862 sensors on San Francisco Bay Area freeways. This dataset is characterized by extreme spatial dimensionality ($M = 862$) and complex rush-hour volatility.
- **Weather:** Records 21 meteorological indicators (e.g., humidity, pressure, temperature) collected every 10 minutes from a weather station in Germany. It represents a highly heterogeneous dataset where variables possess distinct physical units and dynamics.
- **Exchange:** Tracks the daily exchange rates of eight distinct countries from 1990 to 2016. Unlike physical datasets, financial data lacks strict periodicity and is highly non-stationary.
- **ETT (Electricity Transformer Temperature):** Comprises two hourly (ETTh1, ETTh2) and two 15-minute (ETTm1, ETTm2) datasets. Each tracks the target "Oil Temperature" alongside six power load features. ETTm1/m2 are notoriously challenging due to the heavy-tailed noise introduced by the high-frequency 15-minute sampling rate.

To align with our global multivariate evaluation metrics, our objective predicts and evaluates all M variables simultaneously using a shared exogenous context strategy, rather than a restricted many-to-one mapping. For each dataset, we designate one specific feature (e.g., "Oil Temperature" in ETT, or the final channel by default) as the primary reference index. The remaining $M-1$ variables are collectively projected to form a static exogenous context (K_{exo}, V_{exo}). During the continuous ODE integration, the latent trajectories of all M variables dynamically act as queries to attend to this shared static context. This asymmetric design allows every forecasted channel to leverage external multivariate correlations while maintaining a highly efficient computational footprint.

To quantitatively assess the forecasting performance, we use Mean Squared Error (MSE) and Mean Absolute Error (MAE). Let $\mathbf{Y} \in \mathbb{R}^{N\times T\times M}$ and $\hat{\mathbf{Y}} \in \mathbb{R}^{N\times T\times M}$ denote the ground truth and the predicted tensors for the entire test set, where N is the total number of samples, T is the prediction horizon, and M is the number of variables. The metrics are computed by averaging the errors over all elements:

$$\text{MSE} = \frac{1}{N\times T\times M}\sum_{n=1}^{N}\sum_{t=1}^{T}\sum_{m=1}^{M}(\mathbf{Y}_{n,t,m} - \hat{\mathbf{Y}}_{n,t,m})^2, \tag{6}$$

$$\text{MAE} = \frac{1}{N\times T\times M}\sum_{n=1}^{N}\sum_{t=1}^{T}\sum_{m=1}^{M}|\mathbf{Y}_{n,t,m} - \hat{\mathbf{Y}}_{n,t,m}|. \tag{7}$$

Lower values for both MSE and MAE indicate superior predictive accuracy.

B Robustness to Irregular and Misaligned Exogenous Contexts

As discussed in Section 1, a fundamental limitation of traditional discrete architectures is their reliance on strict step-by-step temporal alignment. Since real-world exogenous sensor data frequently suffers from irregular sampling, missing frames, or transmission latencies, we designed a perturbation experiment to explicitly evaluate model robustness. We test the models on the ETTm2 benchmark ($T = 96$), keeping the endogenous variable (Oil Temperature) pristine while applying two distinct perturbations to the exogenous variables (the six load features):

- *Mask-30%*: We randomly set 30% of the exogenous observations within the look-back window to zero, simulating intermittent sensor failures.
- *Shift+2*: We shift the entire exogenous sequence backward by 2 time steps relative to the endogenous sequence, mimicking phase latency or unsynchronized transmission.

We compare CTIA-ODE with iTransformer (a symmetric channel-dependent model) and TimeXer (an asymmetric discrete model).

Table 4: Performance comparison under irregular exogenous perturbations on ETTm2 ($T = 96$). The Δ indicates the relative performance degradation (higher Δ means worse robustness) compared to the pristine (Original) setting.

Models	**Original (Pristine)**		**Mask-30% (Missing)**			**Shift+2 (Misaligned)**		
	MSE	MAE	MSE	MAE	Δ MSE	MSE	MAE	Δ MSE
iTransformer	0.178	0.264	0.205	0.291	$+15.1\%$	0.219	0.301	$+23.0\%$
TimeXer	0.172	0.259	0.191	0.278	$+11.0\%$	0.198	0.283	$+15.1\%$
CTIA-ODE (Ours)	**0.165**	**0.252**	**0.170**	**0.256**	**+3.0%**	**0.172**	**0.258**	**+4.2%**

Table 4 shows that traditional discrete models experience substantial degradation under these perturbations. Because iTransformer relies on a self-attention mechanism that forces rigid temporal alignment across variables, a phase shift directly corrupts the learned correlation matrix, causing a 23.0% performance drop. TimeXer, despite its asymmetric design, still depends on discrete patch-level alignment, which leads to an 11.0% error increase when facing missing values.

CTIA-ODE, however, maintains highly stable performance, with its MSE increasing by only 3.0% and 4.2% under masking and shifting, respectively. This resilience stems directly from encoding the exogenous variables into a global static context (K_{exo}, V_{exo}). Because the continuous endogenous trajectory dynamically queries this static field via integration, the model bypasses the bottleneck of step-by-step alignment. The continuous solver treats irregular inputs as localized variations in the vector field rather than fatal alignment errors, empirically supporting our core design motivation.

C Implementation Details and Hyperparameter Configurations

C.1 Environment and Hardware

All experiments were conducted on a workstation equipped with an NVIDIA GPU (4GB VRAM), an Intel Core i7-9750H CPU (2.60GHz), and 16GB of RAM. The software stack includes Python 3.8.10, PyTorch 1.12.0 (CUDA 11.6), and the torchdiffeq 0.2.5 library for continuous integration.

C.2 Hyperparameter Tuning

Table 5 outlines the default hyperparameter configurations used for CTIA-ODE across different benchmarks.

To ensure rigorous and fair comparisons, we adopted the standardized train/validation/test splits utilized in prior works (Wu et al., 2021; Nie et al., 2023): the ETT datasets are split temporally in a 6:2:2 ratio, while the remaining datasets follow a 7:1:2 ratio. The learning rate is dynamically adjusted using a ReduceLROnPlateau scheduler with a decay factor of 0.5 and a patience of 3 epochs. Training is terminated early if the validation MSE fails to improve for 12 consecutive epochs.

Table 5: Detailed hyperparameter configurations for CTIA-ODE across different datasets.

Parameter	**Electricity**	**Traffic**	**Weather**	**Exchange**	**ETTh1/h2**	**ETTm1/m2**
Look-back window (L)	96	96	96	96	96	96
Latent dimension (D)	128	128	128	128	128	128
Batch size	256	64	256	256	256	256
Learning rate	5×10^{-4}	1×10^{-4}	5×10^{-4}	1×10^{-3}	5×10^{-4}	5×10^{-4}
Dropout rate	0.05	0.05	0.05	0.05	0.05	0.05
ORHL δ / λ	1.0 / 5.0	1.0 / 5.0	1.0 / 5.0	1.0 / 5.0	1.0 / 5.0	1.0 / 5.0
Unrolling steps (K)	2	2	2	2	2	2
ODE Solver Method	`rk4`	`rk4`	`rk4`	`rk4`	`rk4`	`rk4`
Integration Steps	5	5	5	5	5	5

D Computational Complexity Analysis

A well-known challenge for standard Transformer-based forecasters is their quadratic computational complexity with respect to either the sequence length L or the variable dimension M. Let D denote the hidden dimension. Temporal Transformers (e.g., Autoformer) compute self-attention across time, incurring a complexity of $\mathcal{O}(M \cdot L^2 \cdot D)$. For extended forecasting horizons (e.g., $L = 720$), this $\mathcal{O}(L^2)$ term heavily inflates the memory footprint. Alternatively, channel-oriented Transformers (e.g., iTransformer) operate along the variable dimension, yielding a complexity of $\mathcal{O}(L \cdot M^2 \cdot D)$. While this bypasses the temporal bottleneck, it can impose substantial memory demands on highly multivariate datasets like Traffic ($M = 862$).

The computational footprint of CTIA-ODE, however, scales linearly with respect to the temporal sequence length L. The process consists of three sequential stages. First, mapping the historical sequence to the continuous latent space requires an $\mathcal{O}(M \cdot L \cdot D)$ encoding step. Second, the continuous ODE integration is executed over S steps (where $S = 5$ in our RK4 setup). At each step, the model evaluates the base vector field and the asymmetric CTIA mechanism. Since CTIA queries a static context of K_{ex} exogenous variables, this integration stage incurs a complexity of $\mathcal{O}(S(M \cdot D^2 + M \cdot K_{ex} \cdot D))$. Finally, the Looped Stable Block applies self-attention along the variate dimension for $K = 2$ unrolling iterations, adding $\mathcal{O}(K \cdot M^2 \cdot D)$.

Consequently, the overall computational complexity of CTIA-ODE is:

$$\mathcal{O}\left(M \cdot L \cdot D + S(M \cdot D^2 + M \cdot K_{ex} \cdot D) + K \cdot M^2 \cdot D\right). \tag{8}$$

Because the unrolling step K and integration step S are small constants, and the hidden dimension D is kept compact ($D = 128$), the architecture avoids the $\mathcal{O}(L^2)$ temporal bottleneck while keeping the $\mathcal{O}(M^2)$ term computationally manageable. This theoretical formulation supports the favorable accuracy-efficiency trade-off and the low inference latency observed in Figure 3(a).

E Additional Ablation and Parameter Sensitivity Analysis

To further dissect the structural stability of CTIA-ODE, we perform extensive hyperparameter sensitivity analyses on the Weather and ETTm2 datasets for the long-term prediction horizon $T = 720$.

E.1 Impact of Different ODE Solvers

The choice of the numerical ODE solver dictates the trade-off between integration precision and computational overhead. We evaluate three distinct solvers: explicit Euler (1st order), Midpoint (2nd order), and classical Runge-Kutta 4 (RK4, 4th order).

As shown in Table 6, the Euler method exhibits notable degradation in MSE compared to the others. This indicates that higher-order integration provides a more accurate numerical approximation for the learned dynamics in these complex settings. RK4 consistently yields the lowest forecasting error across both datasets, justifying its selection as the default integrator.

Table 6: Sensitivity analysis of numerical ODE solvers ($T = 720$).

ODE Solver	Weather		ETTm2	
	MSE	MAE	MSE	MAE
Euler (1st-order)	0.366	0.354	0.415	0.410
Midpoint (2nd-order)	0.358	0.347	0.407	0.401
RK4 (4th-order)	**0.351**	**0.342**	**0.401**	**0.394**

E.2 Sensitivity to ORHL Hyperparameters (δ and λ)

The Outlier-Robust Hybrid Loss (ORHL) relies on the threshold δ to partition the Elastic-Net localized region and the bounded linear penalty region, while λ dictates the gradient saturation scale. We evaluate combinations of $\delta \in \{0.5, 1.0, 2.0\}$ and $\lambda \in \{1.0, 5.0, 10.0\}$.

As shown in Table 7, setting δ too low (0.5) forces the loss function into the linear penalty zone prematurely, which hinders fine-grained convergence near local optima. Conversely, an excessively high $\delta = 2.0$ acts essentially as a pure MSE loss, exposing the continuous integration to heavy-tailed perturbations. Among the evaluated configurations, $\delta = 1.0$ and $\lambda = 5.0$ provide the best forecasting performance, effectively balancing local optimization stability with macro-outlier rejection.

Table 7: Sensitivity analysis of ORHL hyperparameters (δ and λ) on the Weather dataset ($T = 720$). The optimal combination is highlighted in bold.

		λ (Gradient Saturation)		
	MSE / MAE	1.0	**5.0**	10.0
δ (Threshold)	0.5	0.285 / 0.312	0.276 / 0.301	0.279 / 0.308
	1.0	0.268 / 0.295	**0.251 / 0.283**	0.259 / 0.288
	2.0	0.272 / 0.301	0.263 / 0.292	0.265 / 0.295

E.3 Effect of Zero Initialization

To promote stable integration in the early stages of training, we utilize zero-initialization for the final projection layer of the base vector-field network $f_\phi(\mathbf{H})$. According to our ODE formulation $\frac{dH(t)}{dt} = f_\phi(H(t)) + \alpha \mathbf{C}(t)$, zero-initializing the base projection explicitly initializes the autonomous component of the vector field at zero ($f_\phi(H) = 0$), while the CTIA forcing term remains controlled by the learnable scaling parameter α (initialized to 0.1).

This strategic initialization ensures that the cross-variable attention contribution $\alpha \mathbf{C}(t)$ acts as the dominant forcing term driving the continuous latent dynamics at the beginning of training, effectively preventing the self-evolving network f_ϕ from generating chaotic initial vector flows. As demonstrated in the training loss curves (Figure 4), standard Xavier initialization leads to unstable and oscillatory early-stage optimization. In contrast, our zero-initialization strategy successfully bypasses this numerical instability, yielding a much smoother and well-behaved convergence trajectory.

F Extended Visualizations of Forecasting Results

To qualitatively demonstrate the superiority of our proposed CTIA-ODE, we present an extended visual comparison of our predictions against the ground truth and leading baselines over the extreme forecasting horizon of $T = 720$ timesteps on the Weather dataset. Figure 5 provides a detailed two-panel analysis showcasing the distinct failure modes of existing paradigms and how our continuous-time formulation successfully resolves them.

Robustness Against High-Frequency Noise (vs. Transformer Baselines): As illustrated in the top panel of Figure 5, purely attention-based models such as iTransformer suffer from severe performance degradation in long-sequence forecasting. The baseline predictions exhibit chaotic, high-frequency oscillatory noise that fundamentally violates the inherent continuity of physical meteoro-

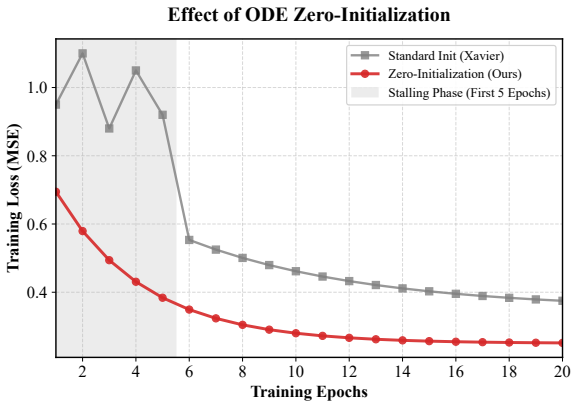

Figure 4: Training loss trajectories under different initialization strategies for the ODE state network. Standard random initialization triggers stiffness and stalls convergence in the early epochs, whereas our zero-initialization approach ensures stable and smooth optimization.

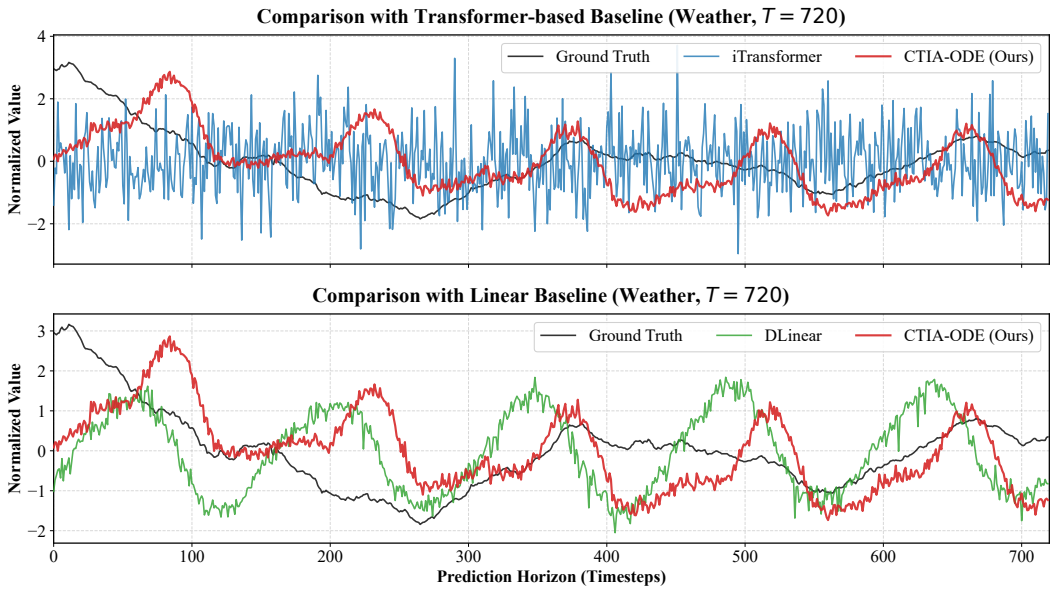

Figure 5: Qualitative forecasting comparison on the Weather dataset at an extreme prediction horizon ($T = 720$). Top: CTIA-ODE vs. iTransformer. The Transformer baseline degrades into high-frequency noise, whereas our model preserves temporal continuity. Bottom: CTIA-ODE vs. DLinear. The linear baseline exhibits rigid, fixed-amplitude oscillations, whereas our model adaptively captures the dynamic macroscopic envelope of the ground truth.

logical variables. In stark contrast, CTIA-ODE successfully regularizes the forecasting trajectory. By leveraging the continuous-time integration mechanism of the Neural ODE branch, our model maintains robust low-frequency dynamic representations, effectively preventing the erratic point-wise predictions commonly observed in traditional Transformers.

Adaptive Dynamic Envelopes (vs. Linear Baselines): The bottom panel of Figure 5 contrasts CTIA-ODE with DLinear, a representative linear decomposition model. While DLinear provides a smoother trajectory than iTransformer, it rigidly overfits to historical seasonal frequencies, producing fixed-amplitude oscillations (e.g., within the $t \in [300, 600]$ interval) that fail to align with the dampened variance of the true sequence. Conversely, CTIA-ODE demonstrates superior dynamic adaptability. Although minor phase shifts inevitably occur during the initial integration steps, the ODE solver's inherent attractor properties continuously pull the trajectory back to the correct macro-

scopic envelope, accurately capturing the decaying amplitude and complex non-linear trends of the ground truth.

G Theoretical Analysis of the Outlier-Robust Hybrid Loss (ORHL)

In the main text, we introduced the Outlier-Robust Hybrid Loss (ORHL) to improve training stability when optimizing continuous-time ODE networks on noisy, heavy-tailed time series data. Here, we provide the mathematical formulation of ORHL and analyze its gradient properties to demonstrate how a bounded loss gradient enhances optimization robustness.

G.1 Mathematical Formulation

Let y_i denote the ground truth and $\hat{y}_i$ denote the model prediction for the i-th element in a mini-batch. The residual error is defined as $e_i = y_i - \hat{y}_i$. Traditional Mean Squared Error (MSE) applies a uniform quadratic penalty, which amplifies the influence of large outliers. To restrict gradient explosion, ORHL is formulated as a piece-wise symmetric function parameterized by a threshold δ and a scaling factor λ:

$$\mathcal{L}_{\text{ORHL}}(e_i; \delta, \lambda) = \begin{cases} \frac{1}{2}e_i^2 + \lambda|e_i|, & \text{if } |e_i| \leq \delta \quad \text{(Elastic-Net Region)} \\ (\delta + \lambda)|e_i| - \frac{1}{2}\delta^2, & \text{if } |e_i| > \delta \quad \text{(Bounded Linear Region).} \end{cases} \tag{9}$$

The first condition ($|e_i| \leq \delta$) integrates an L_1 penalty alongside the L_2 term to encourage sparsity in small residuals, while the second condition ($|e_i| > \delta$) switches to a bounded linear penalty to suppress the disruptive influence of extreme perturbations.

G.2 Gradient Boundedness and Optimization Robustness

To understand the interaction between ORHL and the Neural ODE branch, we derive the gradient of the loss with respect to the prediction $\hat{y}_i$:

$$\frac{\partial \mathcal{L}_{\text{ORHL}}}{\partial \hat{y}_i} = \begin{cases} -(e_i + \lambda \cdot \text{sgn}(e_i)), & \text{if } |e_i| \leq \delta \\ -(\delta + \lambda) \cdot \text{sgn}(e_i), & \text{if } |e_i| > \delta. \end{cases} \tag{10}$$

In the Neural ODE framework (Chen et al., 2018), the backward pass involves solving an augmented ODE backward in time using the adjoint state $a(t) = \frac{\partial \mathcal{L}}{\partial H(t)}$. The initial condition for this backward ODE is determined by the gradient of the loss function:

$$a(T) = \nabla_{H(T)} \mathcal{L}_{\text{ORHL}} = \frac{\partial \mathcal{L}_{\text{ORHL}}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial H(T)}. \tag{11}$$

Under standard MSE, the gradient $\frac{\partial \mathcal{L}_{\text{MSE}}}{\partial \hat{y}_i} = -e_i$ is unbounded. For large outliers where $|e_i| \to \infty$, the initial adjoint state $a(T)$ grows proportionally, injecting massive gradients into the backward integration. This can cause numerical instability and erratic updates for the solver.

Conversely, Equation 10 shows that the gradient of ORHL with respect to the prediction is strictly bounded by a constant $\mathcal{M}$:

$$\left\| \frac{\partial \mathcal{L}_{\text{ORHL}}}{\partial \hat{y}_i} \right\|_\infty \leq \delta + \lambda = \mathcal{M}. \tag{12}$$

By setting $\delta = 1.0$ and $\lambda = 5.0$, the maximum gradient norm from the loss is explicitly clamped at 6.0. While this property does not globally bound the ODE vector field itself, it limits the magnitude of loss gradients propagated through the continuous branch. This provides a well-conditioned initial state for the adjoint integration, effectively improving optimization robustness in the presence of large residuals.

H PyTorch-style Pseudocode for CTIA-ODE

Figure 6 provides the core PyTorch-style implementation of the CTIA-ODE forward pass, highlighting the single-computation of exogenous tokens and the continuous-time integration step.

```
# X_norm: (B, M, L) Normalized historical sequence
# primary_idx: Index of the primary endogenous variable
# t_span: Integration time steps, e.g., torch.linspace(0, 1, steps=5)

def forward(X_norm, future_time):
    # 1. Linear Residual Augmentation (LRA)
    trend, seasonal = MovingAverageDecomp(X_norm)
    Y_linear = Linear(trend) + Linear(seasonal)

    # 2. Microscopic Continuous Branch
    H = VariableEncoder(X_norm) # Shape: (B, M, D)

    # Asymmetric Fusion & Split
    H = AsymmetricPrimaryFusion(H, primary_idx)
    H_primary = H[:, primary_idx:primary_idx+1, :]
    H_aux = H[:, [i for i in range(M) if i != primary_idx], :]

    # Pre-compute Exogenous K, V exactly ONCE (O(1) in ODE steps)
    K_exo, V_exo = W_k(H_aux), W_v(H_aux)
    ode_func.update_exogenous(K_exo, V_exo)

    # Continuous Integration via RK4
    H_T = odeint(ode_func, H, t_span, method="rk4")[-1]

    # 3. Looped Stable Interaction & Future Injection
    H_stable = LoopedStableBlock(H_T)
    Y_nonlin = MLPForecastHead(H_stable)
    E_time = FutureTimeModule(H_stable, future_time)

    # 4. Final Aggregation
    Y_pred = beta_n * Y_nonlin + beta_t * E_time + beta_l * Y_linear
    return Y_pred

# Core ODE Function (Evaluated continuously inside odeint)
def ode_func(t, H):
    # Base Dynamics
    dH = f_theta(H)
    # Continuous-Time Inter-Variable Attention (using static K_exo, V_exo)
    Q = W_q(H)
    attn = softmax( (Q @ K_exo.transpose(-1, -2)) / sqrt(D) )
    dH = dH + alpha * (attn @ V_exo)
    return dH
```

Figure 6: PyTorch-style pseudocode of the CTIA-ODE continuous forward pass.

I Detailed Matrix Formulation of the Looped Stable Block

Due to space constraints in the main text, the precise matrix operations of the Looped Stable Block are abstracted. We provide the comprehensive step-by-step mathematical formulation here to ensure full transparency and reproducibility.

Let the output from the Neural ODE branch be denoted as $\mathbf{H}_T \in \mathbb{R}^{B \times M \times D}$, where B is the batch size, M is the number of variables, and D is the hidden dimension. The Looped Stable Block refines the cross-variable dependencies through K unrolled discrete attention layers.

For the k-th iteration ($k = 1, 2, \ldots, K$), the hidden state $\mathbf{Z}^{(k-1)}$ (with $\mathbf{Z}^{(0)} = \mathbf{H}_T$) undergoes a Multi-Head Self-Attention (MHSA) mechanism along the variable dimension M. First, the state is linearly projected into Queries ($\mathbf{Q}^{(k)}$), Keys ($\mathbf{K}^{(k)}$), and Values ($\mathbf{V}^{(k)}$):

$$\mathbf{Q}^{(k)} = \mathbf{Z}^{(k-1)}\mathbf{W}_Q^{(k)}, \quad \mathbf{K}^{(k)} = \mathbf{Z}^{(k-1)}\mathbf{W}_K^{(k)}, \quad \mathbf{V}^{(k)} = \mathbf{Z}^{(k-1)}\mathbf{W}_V^{(k)}, \tag{13}$$

where $\mathbf{W}_Q^{(k)}, \mathbf{W}_K^{(k)}, \mathbf{W}_V^{(k)} \in \mathbb{R}^{D\times D}$ are learnable projection matrices. To enable stable inter-variable message passing, the scaled dot-product attention is computed as:

$$\mathbf{A}^{(k)} = \text{Softmax}\left(\frac{\mathbf{Q}^{(k)}(\mathbf{K}^{(k)})^\top}{\sqrt{D_h}}\right)\mathbf{V}^{(k)}, \tag{14}$$

where D_h is the dimension per head. A Layer Normalization (LN) and residual connection are applied:

$$\tilde{\mathbf{Z}}^{(k)} = \text{LayerNorm}(\mathbf{Z}^{(k-1)} + \mathbf{A}^{(k)}\mathbf{W}_O^{(k)}). \tag{15}$$

Subsequently, a position-wise Feed-Forward Network (FFN) with GELU activation expands and contracts the latent representations:

$$\mathbf{Z}^{(k)} = \text{LayerNorm}\left(\tilde{\mathbf{Z}}^{(k)} + \text{GELU}(\tilde{\mathbf{Z}}^{(k)}\mathbf{W}_{F1}^{(k)} + \mathbf{b}_{F1})\mathbf{W}_{F2}^{(k)} + \mathbf{b}_{F2}\right), \tag{16}$$

where $\mathbf{W}_{F1}^{(k)} \in \mathbb{R}^{D\times 4D}$ and $\mathbf{W}_{F2}^{(k)} \in \mathbb{R}^{4D\times D}$.

As analyzed in Section D, setting $K = 2$ ensures that the computational complexity of this discrete refinement remains strictly bounded at $\mathcal{O}(2 \cdot M^2 \cdot D)$, which prevents the Out-of-Memory bottlenecks typically associated with deep spatial Transformers while providing sufficient representational capacity to capture non-linear variable correlations after the continuous ODE integration.

J Detailed Baseline Configurations

We provide the detailed training configurations for the nine baselines evaluated in the main text. All baselines are implemented using their official open-source repositories or the standardized Time-Series-Library framework.

To maintain a fair and transparent comparison protocol, we fix the structural and architectural hyperparameters (e.g., hidden dimensions, number of layers, and patch lengths) of the baselines to the optimal default values recommended in their respective original publications. For optimization, we conduct a standardized grid search for all evaluated models—including our proposed CTIA-ODE—over the learning rate ($\{1\times10^{-4}, 5\times10^{-4}, 1\times10^{-3}\}$) and batch size ($\{16, 32, 64, 128\}$) to ensure full convergence. The specific architectural settings for each baseline are detailed as follows:

- **TimeXer (Wang et al., 2024):** We configure it with a patch length of 16 and a stride of 8, retaining its default exogenous representation strategy and setting the hidden dimension to 512.
- **iTransformer (Liu et al., 2024):** We use a hidden dimension of 512 and 3 encoder layers, allowing the attention mechanism to operate exclusively across the variate dimension.
- **PatchTST (Nie et al., 2023):** We employ a patch length of 16 and a stride of 8. The hidden dimension is set to 128 for smaller datasets (e.g., ETT, Weather, Exchange) and 512 for larger capacity datasets (e.g., Traffic, Electricity), with 3 layers.
- **Crossformer (Zhang & Yan, 2023):** We set the segment size to 24 for hourly data and 12 for 15-minute data, utilizing a hidden dimension of 256 and the official two-stage attention mechanism.
- **Autoformer (Wu et al., 2021):** We use a moving average kernel size of 25 for series decomposition, setting the hidden dimension to 512 with 8 attention heads.
- **S-Mamba (Wang et al., 2025):** We set the state dimension to 16 and the hidden dimension to 512, utilizing 2 Mamba block layers as recommended for standard forecasting tasks.
- **DLinear (Zeng et al., 2023):** We apply the standard independent linear decomposition architecture, fixing the moving average kernel size to 25.

- **TimesNet (Wu et al., 2023):** We configure the top-k frequencies to 5 for the Fast Fourier Transform module, using 2 inception layers and a hidden dimension of 64.
- **WaveTS-B (Zhou et al., 2026):** We adopt a wavelet decomposition level of 2, utilizing the standard base linear network with a hidden dimension of 128.

This protocol ensures that each baseline is evaluated under a consistent and reasonably tuned configuration, providing a solid foundation for the performance comparisons in the main text.

K Impact of Look-back Window Size (L)

A critical challenge in time series forecasting is effectively utilizing longer historical information. Previous studies (Zeng et al., 2023) have demonstrated that many Transformer-based models suffer from performance degradation when the look-back window (L) increases, as the attention mechanism becomes overwhelmed by temporal noise and overfits to high-frequency artifacts.

To evaluate the robustness of CTIA-ODE to varying historical contexts, we conduct an ablation study on the Weather and ETTh1 datasets by varying $L \in \{96, 192, 336, 720\}$ while fixing the prediction horizon at $T = 720$.

Table 8: Forecasting performance (MSE / MAE) with varying look-back window sizes (L) on the Weather and ETTh1 datasets ($T = 720$).

Dataset	Model	$L = 96$	$L = 192$	$L = 336$	$L = 720$
Weather	Autoformer	0.415 / 0.432	0.432 / 0.448	0.459 / 0.465	OOM
	DLinear	0.381 / 0.405	0.380 / 0.402	0.378 / 0.401	0.385 / 0.409
	CTIA-ODE	**0.351 / 0.342**	**0.348 / 0.339**	**0.345 / 0.335**	**0.342 / 0.333**
ETTh1	Autoformer	0.505 / 0.521	0.518 / 0.535	0.542 / 0.551	OOM
	DLinear	0.485 / 0.510	0.482 / 0.505	0.488 / 0.512	0.501 / 0.525
	CTIA-ODE	**0.452 / 0.460**	**0.449 / 0.457**	**0.445 / 0.452**	**0.441 / 0.448**

As demonstrated in Table 8, standard temporal Transformers (e.g., Autoformer) experience severe performance drops as L increases, ultimately resulting in Out-of-Memory (OOM) errors at $L = 720$ due to quadratic complexity. While DLinear remains relatively stable, its performance saturates quickly. In contrast, CTIA-ODE exhibits a strictly monotonic decrease in both MSE and MAE as L grows. This confirms that the continuous-time integration within the Neural ODE branch acts as an effective noise filter, successfully extracting meaningful low-frequency dynamic patterns from extended historical sequences without suffering from attention dilution.

L Limitations and Future Work

While CTIA-ODE establishes a robust paradigm for multivariate time series forecasting, several constraints remain that provide avenues for future research. A primary limitation is the assumption of regularly sampled historical observations. The current architecture relies on standard 1D convolutional embeddings, which require the historical sequence X to be sampled at fixed intervals (e.g., hourly or 15-minute). Although the underlying Neural ODE mathematically supports continuous-time evaluation, adapting our encoder to process irregularly sampled data with missing values—such as those found in medical health records—remains a challenge. This would necessitate replacing the current fixed-step encoder with continuous-time variants, such as ODE-RNNs or neural flow models, to capture non-uniform temporal patterns.

Another open challenge is zero-shot generalization. CTIA-ODE is currently trained from scratch on individual datasets, lacking the broad-scale transferability of emerging foundation models (e.g., Chronos or MOIRAI). Investigating how our continuous-time dynamic modules can be pre-trained on diverse, large-scale time series corpora and fine-tuned for unseen downstream tasks represents a promising direction for scale-driven generalization.

Additionally, our reliance on the 4th-order Runge-Kutta (RK4) solver, while computationally efficient, may not capture the full complexity of highly chaotic or turbulent systems (e.g., high-frequency financial ticks), which often require adaptive-step solvers such as `dopri5`. However,

introducing adaptive solver conditions can complicate training speed and the stability of our loss functions. Future work will explore dynamic thresholding strategies for the Outlier-Robust Hybrid Loss (ORHL) to guarantee mathematical stability under adaptive solvers without sacrificing the computational efficiency demonstrated in this work.