SPECTRAL-TEMPORAL DECOUPLED SPARSE ATTACK ON PARAMETER-EFFICIENT TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in time series forecasting (TSF) have embraced architectural minimalism and parameter efficiency, but still achieved competitive accuracy. However, extreme structural sparsity and drastically reduced parameter spaces are vulnerable to adversarial perturbations. In this work, we propose a spectral-temporal decoupled sparse attack on parameter-efficient (PE) TSF models, which simultaneously disrupts localized morphological shapes and dominant periodic trajectories. Extensive experiments show that our method successfully sabotages five state-of-the-art PE TSF models for a small ℓ_∞ perturbation budget $\epsilon = 0.1$ (SNR≈ 21.52dB) without any structural or parameter queries, even under six representative adversarial defense measures. It can be applied to non-query attacks on remote edge-devices, stress testing in critical infrastructure domains, and structural diagnosing on PE edge-device deployment.

1 INTRODUCTION

Time series forecasting (TSF) is an important decision-making methodology across numerous critical domains, such as energy grid management (Ma et al., 2025; Zhou et al., 2023), macroeconomic planning (Zhao & Shen, 2025; Qiu et al., 2025), climate risk assessment (Dama et al., 2025; Majeedi et al., 2025), and language models (Jin et al., 2024; Ansari et al., 2024). Recently, this methodology has progressed toward architectural minimalism and parameter efficiency (PE), in order to relieve the computational burden and the overfitting problem. For example, FITS (Xu et al., 2024), SparseTSF (Lin et al., 2024), and TimeBase (Huang et al., 2025) have aggressively reduced the entire network to lightweight linear and patching operators with minimal parameters. iTransformer (Liu et al., 2024) projects variable dimensions independently to avoid quadratic complexity, while LSINet (Zhang et al., 2025a) enforces Bernoulli-gate sparse interactions. To achieve competitive forecasting accuracy within such tight parameter constraints, these methods rely on two key structural mechanisms: **1. Time-domain low-rank mapping:** it folds one-dimensional temporal sequences into low-rank spatial matrices via segmented block-remapping to extract localized trends. **2. Frequency-domain low-pass filtering:** it truncates high-frequency spectral components via fast Fourier transforms (FFT) to focus specifically on dominant seasonal periodicities.

However, extreme parameter efficiency and structural sparsity may expose such TSF models to carefully crafted adversarial attacks. Because these models concentrate on a restricted subspace of low-frequency periodicities and localized low-rank temporal structures, a small and specialized perturbation injected directly into this subspace may trigger serious cross-period forecasting failures. While universal adversarial attacks like Projected Gradient Descent (PGD, Madry et al. 2018; Croce & Hein 2020) and C&W (Carlini & Wagner, 2017) have been applied to time series (Ding et al., 2023; Liu et al., 2023), specialized attacks targeting PE TSF with no query to and minimal knowledge of such models remain sparse.

In this paper, we **reveal a critical vulnerability** of PE TSF models that **cautions** the community against optimistic assumptions of **architectural minimalism and parameter efficiency**. We propose a Spectral-Temporal Decoupled Sparse Attack (**STDSA**) on PE TSF models with no query to and minimal knowledge of them. It decouples the input time series into a dual-pathway architecture, which targets the above-mentioned two key structural mechanisms. The first pathway handles

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.

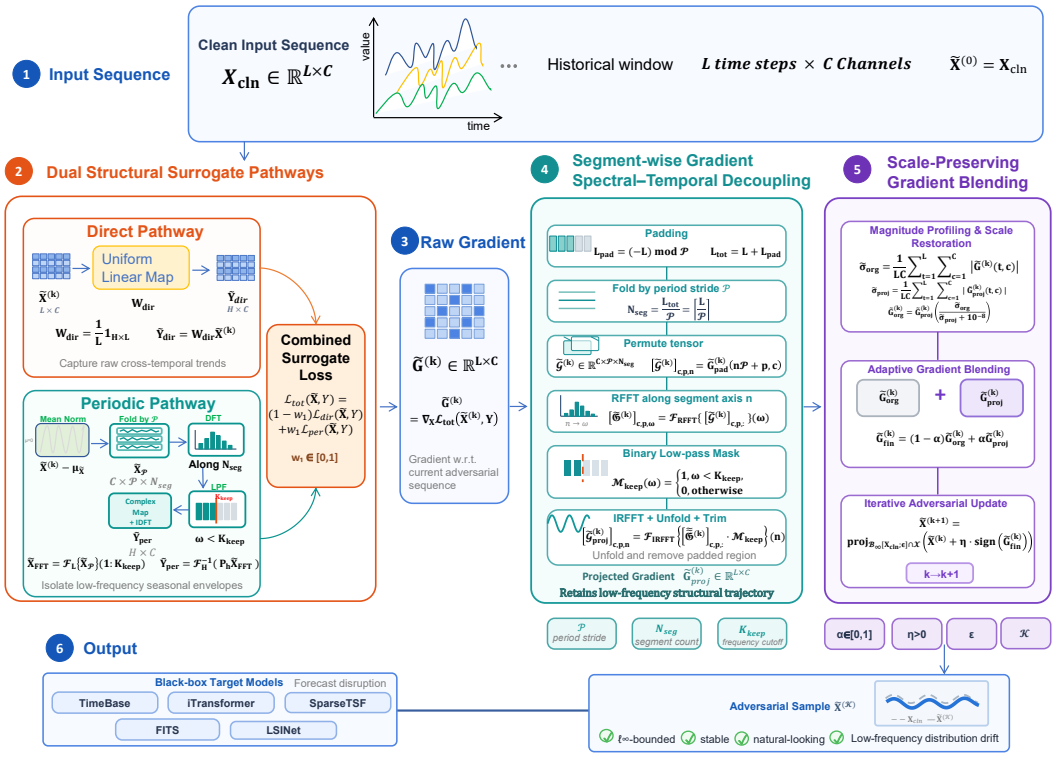

Figure 1: Attack scheme of the proposed STDSA. Its key idea is to build the loss with two structural pathways, project raw gradients onto a low-frequency and segment-wise spectral space, blend the projected gradients, and update the adversarial sample under the ℓ_∞ budget.

temporal folding via low-rank cross-segment linear projections, while the second pathway isolates global trajectories within a truncated low-pass complex frequency space by a low-pass filter (LPF). By optimizing against this joint spectral-temporal bottleneck, STDSA generates imperceptible and high Signal-to-Noise-Ratio (SNR) adversarial samples that bypass the inherent defenses and disrupt the core predictive manifolds of these models. Our main contributions can be summarized as follows:

1. We feature a segment-wise spectral-temporal decoupling mechanism that reshapes gradients based on localized periodic cycles and applies a Real Fast Fourier Transform (RFFT) to mask high-frequency bins, which forces adversarial perturbations to appear as smooth natural-looking low-frequency distribution drifts.

2. We ensemble structurally complementary backbones for time-domain trend, seasonal decomposition, and sub-segment frequency mapping, which generates a hybrid gradient that enhances extendability to attention-heavy target architectures.

3. We use a scale-preserving adaptive gradient blending scheme that dynamically rescales the filtered tensors and linearly interpolates them with the raw high-variance time-domain gradients, which not only ensures a stable optimization path but also retains enough fine-grained localized gradient energy to navigate complex black-box loss landscapes.

STDSA mainly exploits two key mechanisms to construct adversarial samples: the simple one-layer linear model that uses weight tying to reduce parameter complexity (Zeng et al., 2023), and the periodicity capture through LPF (Ma et al., 2026). The attacker neither needs to access the target PE TSF model nor has knowledge on it. He/she just need to inject an imperceptible perturbation into a clean sample, and then finishes the attack (Figure 1). Once the target PE TSF model processes the adversarial sample, its predicting mean squared error (MSE) rises significantly, which makes it ineffective (Figure 2). STDSA successfully disrupts five state-of-the-art (SOTA) PE TSF models: TimeBase (Huang et al., 2025), FITS (Xu et al., 2024), iTransformer (Liu et al., 2024), SparseTSF (Lin et al., 2024), and LSINet (Zhang et al., 2025a), even under six typical defense methods: randomized smoothing (Cohen et al., 2019), MSE consistency regularization (Zhang et al., 2019), Total

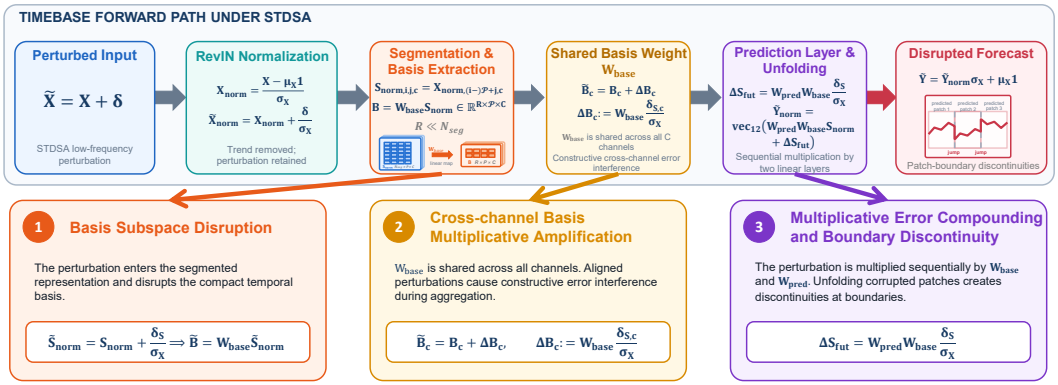

Figure 2: Attack chain of the proposed STDSA on a PE TSF model, TimeBase. Low-frequency perturbations propagate through shared temporal bases and aggregate across different channels. Structured perturbations propagate through shared basis extraction and sequential linear projections, which produces amplified forecasting errors and patch-boundary discontinuities.

Variation (TV) regularization (Lai & Wang, 2024; Wang et al., 2025; Lai et al., 2026a;b), Kullback-Leibler (KL) regularization (Goldblum et al., 2020; Zhang et al., 2025b), adversarial training (AT, Xie et al. 2019; Shafahi et al. 2019), and a combined defense including AT, MSE consistency, and TV. This indicates that the underlying PE mechanisms like time-domain low-rank mapping and frequency-domain LPF are vulnerable to the proposed structural attack.

STDSA can be applied to at least three non-query practical scenarios:

1. It can be deployed as a stealthy non-query evasion mechanism against remote edge-devices, which mimics natural low-frequency environmental drifts or seasonal fluctuations.

2. It can serve as a an automated non-query stress testing engine in critical infrastructure domains, such as smart grid load forecasting, macroeconomic tracking, and algorithmic trading. Such stress testing examines whether the target system can withstand structural frequency-domain disruptions before going live, in order to avoid financial losses or grid failures.

3. It can be used as a structural diagnosing tool on PE edge-device deployment, which helps engineers identify what structural information is lost during compression and where the defensive capacity of the target system collapses.

2 Preliminaries

2.1 Parameter-efficient Time Series Forecasting

Let $X \in \mathbb{R}^{L\times C}$ be the historical lookback sequence over a window length L across C variables, and let $Y \in \mathbb{R}^{H\times C}$ be the ground-truth future sequence over a forecasting horizon H. Traditional deep long-term TSF architectures parameterize a dense mapping $f_\theta : \mathbb{R}^{L\times C} \to \mathbb{R}^{H\times C}, Y = f_\theta(X)$ via multi-layered tokenization or attention patterns. To reduce the parameter scale, PE TSF frameworks enforce structural inductive compacting.

Taking FITS as a representative example, it converts time-domain lookback sequences into the frequency domain by the Discrete Fourier Transform (DFT) operator $\mathcal{F}_L$ with respect to (w.r.t.) the lookback dimensionality L. This yields the spectral representation $Z = \mathcal{F}_L\{X\} \in \mathbb{C}^{K\times C}$ where $K = \lfloor L/2 \rfloor + 1$. Recognizing that real-world macro-dynamics are mostly bounded in a low-frequency envelope, FITS truncates high-frequency spectra via an LPF matrix $W_{LPF} \in \{0,1\}^{K\times C}$ where the first K_{cut} ($K_{cut} \ll K$) rows and the rest are set to 1 and 0, respectively. The forecasting trajectory is optimized via a compact complex linear map $W_f \in \mathbb{C}^{K_{high}\times K}$ (with $K_{high} = \lfloor H/2 \rfloor + 1$) such that:

$$\hat{Y} = \mathcal{F}_H^{-1}\left(\mathcal{P}_H\left(W_f\left(W_{LPF} \odot \mathcal{F}_L\{X\}\right)\right)\right), \tag{1}$$

where $\odot$ denotes the Hadamard product, $\mathcal{P}_H$ denotes a zero-padding transformation that extends the tensor back to the horizon spectrum boundary, and $\mathcal{F}_H^{-1}$ notes the inverse DFT (IDFT) w.r.t. horizontal dimensionality H.

On the other hand, methods like SparseTSF achieve parameter sparsity by executing downsampling mechanics directly in the temporal domain. Given an established period stride $\mathscr{P}$, the historical trace is sliced into cross-period sub-sequences:

$$X_{[p]} = [X_p; X_{p+\mathscr{P}}; X_{p+2\mathscr{P}}; \dots], \tag{2}$$

where X_p denotes the historical lookback at the p-th timestamp. This regularizes multi-step lookbacks into periodic lattices. Similarly, other methods like TimeBase, iTransformer, and LSINet exploit independent variable projections or Bernoulli-gate sparse interactions to maintain accuracy within tight constraints.

2.2 Adversarial Attack on Temporal Sequences

Adversarial attacks targeting time-series models seek to find an additive perturbation matrix $\delta \in \mathbb{R}^{L\times C}$ that maximizes the prediction loss subject to a budget defined within a closed $\ell_{\mathfrak{p}}$-norm ball $\mathcal{B}_{\mathfrak{p}}[0;\epsilon]$. Note that in the literature of adversarial attack, the **$\ell_{\mathfrak{p}}$-norm of a matrix corresponds to the $\ell_{\mathfrak{p}}$-norm of its vectorization**. This optimization can be generalized as:

$$\max_{\|\delta\|_{\mathfrak{p}}\leqslant\epsilon} \mathcal{L}(f_\theta(X+\delta), Y), \tag{3}$$

where $\mathcal{L}$ denotes the forecast loss function (typically Mean Squared Error, MSE). Conventional gradient-based white-box attacks like the Fast Gradient Sign Method (FGSM, Goodfellow et al. 2015) and Projected Gradient Descent (PGD, Madry et al. 2018; Croce & Hein 2020) iteratively push the perturbation along the directional sign of the input loss gradient:

$$\tilde{X}^{(k+1)} = \mathrm{proj}_{\mathcal{B}_{\mathfrak{p}}[X;\epsilon]}(\tilde{X}^{(k)} + \eta \cdot \mathrm{sign}(\nabla_X \mathcal{L}(f_\theta(\tilde{X}^{(k)}), Y))), \tag{4}$$

where $\mathrm{proj}_{\mathcal{B}_{\mathfrak{p}}[X;\epsilon]}$ denotes the orthogonal projection operator onto the closed $\ell_{\mathfrak{p}}$-norm ball, η represents the gradient step size, and $\tilde{X}^{(k)} := X + \delta^{(k)}$ denotes the perturbed lookback sequence at the k-th attack iteration. In the rest of this paper, we add a **tilde** over a variable to denote that it is a **perturbed version**. Another representative method is the C&W attack (Carlini & Wagner, 2017), which solves an optimization problem targeting log-likelihood boundaries.

However, these attack schemes are not necessarily effective in PE TSF models, because such standard local temporal variations may be overwhelmed by high-frequency and noisy variations, or relieved by the filtering operator. For example, adversarial energy concentrated outside the K_{cut} threshold may be eliminated by the filtering operator: $W_{LPF} \odot \mathcal{F}_L\{\delta\} \approx 0$, which may degrade standard gradient attack schemes.

3 Methodology

We observe that although quite a few PE TSF models have investigated smooth transformations or temporally invariant maps to preserve local sample consistency, they remain incapable of independently decoupling the global spectral landscape from localized temporal properties. This gap exposes a critical structural vulnerability that the proposed STDSA attack can exploit (see Figure 1).

3.1 Two Structural Pathways: Direct and Periodic

STDSA blends the loss landscapes of two distinct architectural operators acting as proxy structural paths: the direct one-layer linear structure and the periodic structure through LPF.

$$\mathcal{L}_{tot}(\tilde{X}, Y) = (1-w_1)\cdot\mathcal{L}_{dir}(\tilde{X}, Y) + w_1 \cdot \mathcal{L}_{per}(\tilde{X}, Y), \tag{5}$$

where $w_1 \in [0,1]$ is a mixing parameter. $\mathcal{L}_{dir}$ and $\mathcal{L}_{per}$ correspond to the MSE computed over the forecasting outputs of the **direct** temporal mapping pathway and the **periodic** frequency pathway, respectively.

The direct pathway instantiates a simple one-layer linear mapping to capture raw cross-temporal trends without channel-wise interference. It maps the sequence length dimension to the prediction horizon via a single linear layer. To enforce parameter minimalism and simulate extreme structural constraints, this pathway incorporates a uniform weight-tying configuration as its baseline structural parameterization:

$$W_{dir} = \frac{1}{L}\mathbb{1}_{H\times L}, \quad \tilde{Y}_{dir} = W_{dir}\tilde{X} \in \mathbb{R}^{H\times C}, \tag{6}$$

where $\mathbb{1}_{H\times L}$ denotes the $H \times L$ matrix of ones.

The periodic pathway captures global periodic trajectories by conducting LPF on sub-segment structural blocks. After applying mean normalization, the input sequence is folded based on a period stride $\mathscr{P}$ into a three-dimensional sub-sequence tensor $\tilde{X}_{\mathscr{P}} \in \mathbb{R}^{C\times\mathscr{P}\times N_{seg}}$, where $N_{seg} = \lceil L/\mathscr{P} \rceil$ tracks the segment count. This pathway isolates low-frequency seasonal envelopes by applying a DFT specifically along the segment dimension, which filters high-frequency variations via an LPF boundary K_{keep} and computes complex linear maps:

$$\tilde{\mathcal{X}}_{FFT} = \mathcal{F}_{N_{seg}}\{\tilde{X}_{\mathscr{P}}\}(1 : K_{keep}), \quad \tilde{Y}_{per} = \mathcal{F}_H^{-1}(\mathcal{P}_H \tilde{\mathcal{X}}_{FFT}). \tag{7}$$

Then we insert $\tilde{Y}_{dir}$ and $\tilde{Y}_{per}$ into $\mathcal{L}_{dir}$ and $\mathcal{L}_{per}$ to calculate the MSE losses, respectively.

3.2 Segment-wise Gradient Spectral-Temporal Decoupling

The initial adversarial sample is the clean counterpart $\tilde{X}^{(0)} := X_{cln}$. At the k-th attack iteration, the raw gradient matrix of the combined surrogate loss w.r.t. the current adversarial sequence is computed as:

$$\tilde{G}^{(k)} = \nabla_X \mathcal{L}_{tot}(\tilde{X}^{(k)}, Y) \in \mathbb{R}^{L\times C}. \tag{8}$$

To ensure that the generated perturbation flows smoothly through the downsampling operators and frequency filters of remote target models, we project $\tilde{G}^{(k)}$ into a decoupled cross-period spectral space.

To achieve block alignment across a uniform period stride $\mathscr{P}$, we pad the temporal dimension of $\tilde{G}^{(k)}$ by a padding length $L_{\text{pad}} = (-L) \bmod \mathscr{P}$. The padded gradient matrix $\tilde{G}^{(k)}_{pad} \in \mathbb{R}^{L_{tot}\times C}$ is folded into a three-dimensional tensor and permuted to separate phase positioning from global progression, where $L_{tot} = L + L_{pad}$:

$$\tilde{\mathcal{G}}^{(k)} \in \mathbb{R}^{C\times\mathscr{P}\times N_{seg}}, \quad [\tilde{\mathcal{G}}^{(k)}]_{c,p,n} = \tilde{G}^{(k)}_{pad}((n-1)\cdot\mathscr{P} + p, c). \tag{9}$$

c, p, n denote the indices of channel (C), period stride ($\mathscr{P}$), and global segment (N_{seg}), respectively. Next, we apply a Real Fast Fourier Transform (RFFT) along the segment axis n to identify the global trajectory frequencies:

$$[\tilde{\mathfrak{G}}^{(k)}]_{c,p,\omega} = \mathcal{F}_{RFFT}\{[\tilde{\mathcal{G}}^{(k)}]_{c,p,\cdot}\}(\omega), \tag{10}$$

where ω represents the cross-period frequency index. High-frequency variations across historical blocks are eliminated by applying a binary LPF mask $\mathcal{M}_{keep}(\omega)$ controlled by the frequency cutoff budget K_{keep}. Then the filtered frequency tensor is transformed back into the temporal segment space via the Inverse RFFT (IRFFT):

$$[\tilde{\mathcal{G}}^{(k)}_{\text{proj}}]_{c,p,n} = \mathcal{F}_{IRFFT}\{[\tilde{\mathfrak{G}}^{(k)}]_{c,p,\cdot} \odot \mathcal{M}_{keep}\}(n), \quad \mathcal{M}_{keep}(\omega) = \begin{cases} 1, & \text{if } \omega \leqslant K_{keep} \\ 0, & \text{otherwise} \end{cases}. \tag{11}$$

Unfolding $\tilde{\mathcal{G}}^{(k)}_{\text{proj}}$ and removing the padded regions yields the structural projected gradient $\tilde{G}^{(k)}_{\text{proj}} \in \mathbb{R}^{L\times C}$.

3.3 Scale-Preserving Gradient Blending

Frequency truncation dissipates gradient energy, which can slow down optimization. To stabilize the optimization path while keeping fine-grained local gradient properties intact, we introduce a scale-preserving restoration and adaptive blending scheme. First, the average absolute magnitude profiles of the raw gradient and the filtered gradient are calculated:

$$\tilde{\sigma}_{org} = \frac{1}{L\cdot C}\sum_{t=1}^{L}\sum_{c=1}^{C}|\tilde{G}^{(k)}(t,c)|, \quad \tilde{\sigma}_{\text{proj}} = \frac{1}{L\cdot C}\sum_{t=1}^{L}\sum_{c=1}^{C}|\tilde{G}^{(k)}_{\text{proj}}(t,c)|. \tag{12}$$

The filtered gradient is dynamically rescaled to match the energy envelope of the original time-domain gradient tensor. Then we execute a linear interpolation between the high-variance raw gradient and the smooth low-frequency trajectory with a spectral blending parameter $\alpha \in [0, 1]$:

$$\tilde{G}^{(k)}_{org} = \tilde{G}^{(k)}_{\text{proj}} \cdot \left(\frac{\tilde{\sigma}_{org}}{\tilde{\sigma}_{\text{proj}} + 10^{-8}}\right), \quad \tilde{G}^{(k)}_{fin} = (1-\alpha)\tilde{G}^{(k)}_{org} + \alpha\tilde{G}^{(k)}_{\text{proj}}. \tag{13}$$

The adversarial sequence for the next step is updated by input optimization (Erhan et al., 2009; Goodfellow et al., 2015; Madry et al., 2018; Gurumurthy et al., 2021; Yu & Gao, 2024), and then projected onto the permitted ℓ_∞ budget boundary:

$$\tilde{X}^{(k+1)} = \text{proj}_{\mathcal{B}_\infty[X_{cln};\epsilon]\cap\mathscr{X}}\left(\tilde{X}^{(k)} + \eta \cdot \text{sign}\left(\tilde{G}_{fin}^{(k)}\right)\right), \tag{14}$$

where $\eta > 0$ denotes the step size, and $\text{proj}_{\mathcal{B}_\infty[X;\epsilon]\cap\mathscr{X}}$ projects the updated sample onto the intersection of the sample domain $\mathscr{X}$ and the closed ball $\mathcal{B}_\infty[X_{cln};\epsilon]$ centered at the clean sample X_{cln}. After a sufficient number $\mathscr{K}$ of attack iterations, the resulted adversarial sample $\tilde{X}^{(\mathscr{K})}$ contains stable, natural-looking, and low-frequency distribution drifts that can disrupt other black-box target models.

3.4 Attack Chain

To illustrate the attack chain of STDSA, we take the SOTA PE TSF model TimeBase as an example (see Figure 2), which models time-series sequences as linear or additive combinations of temporal base functions. It projects historical lookback windows directly into future forecast horizons through channel-independent linear mappings or base decomposition. It uses Reversible Instance Normalization (RevIN, Kim et al. 2022) to remove non-stationary trend shifts prior to applying linear transformation layers:

$$X_{norm} = \frac{X - \mu_X \mathbb{1}}{\sigma_X}, \quad \hat{Y}_{norm} = W_{TB} X_{norm}, \quad \hat{Y} = \hat{Y}_{norm}\sigma_X + \mu_X \mathbb{1}, \tag{15}$$

where μ_X and σ_X are the temporal mean and standard deviation (STD) of X, and W_{TB} is the learned weight matrix for TimeBase. It involves several vulnerabilities that can be exploited by STDSA:

Basis subspace disruption. Rather than forecasting point-by-point, TimeBase divides the normalized input $X_{norm} \in \mathbb{R}^{L\times C}$ into N_{seg} non-overlapping segments of length $\mathscr{P}$. For simplicity of presentation, we assume $L = N_{seg} \cdot \mathscr{P}$ and $H = N'_{seg} \cdot \mathscr{P}$. Then it extracts a compact matrix of R fundamental basis temporal components $B = W_{base} S_{norm} \in \mathbb{R}^{R\times\mathscr{P}\times C}$ (where $R \ll N_{seg}$) via a linear mapping layer W_{base}, where $S_{norm} \in \mathbb{R}^{N_{seg}\times\mathscr{P}\times C}$ represents the segmented matrix of the lookback window:

$$S_{norm,i,j,c} = X_{norm,(i-1)\cdot\mathscr{P}+j,c}, \quad \text{for } i \in [N_{seg}] := \{1,\ldots,N_{seg}\},\ j \in [\mathscr{P}], c \in [C]. \tag{16}$$

When X is perturbed to be $\tilde{X} = X + \delta$, the perturbed $\tilde{X}_{norm} = X_{norm} + \frac{\delta}{\sigma_X}$. Since $\tilde{S}_{norm}$ is a projected matrix of $\tilde{X}_{norm}$, we have $\tilde{S}_{norm} = S_{norm} + \frac{\delta_S}{\sigma_X}$, where δ_S is the corresponding projected matrix of δ. Then the basis is perturbed to be: $\tilde{B} = W_{base}(S_{norm} + \frac{\delta_S}{\sigma_X})$. Once the basis is disrupted, the **downstream pipelines and other mechanisms of TimeBase will also be disrupted**.

Cross-channel basis multiplicative amplification. In the standard channel-independent TimeBase, the basis extraction weight $W_{base} \in \mathbb{R}^{R\times N_{seg}}$ is **shared across all** C **channels** to reduce parameters. When the adversarial perturbation $\delta_{S,c}$ reaches channel c:

$$\tilde{B}_c = W_{base}(S_{norm,c} + \frac{\delta_{S,c}}{\sigma_X}) = B_c + \Delta B_c, \quad \Delta B_c := W_{\text{base}}\frac{\delta_{S,c}}{\sigma_X}. \tag{17}$$

If $\delta_{S,c}$ aligns across different channels, it will cause **constructive error interference** during global forecast aggregation, which can be **amplified** by the shared W_{base}.

Multiplicative error compounding and boundary discontinuity. To project future horizons, TimeBase applies a prediction layer W_{pred} to the perturbed basis $\tilde{B}$, which produces predicted patches. Then it is unfolded back into a continuous timeline $\tilde{Y}$:

$$\Delta S_{fut} = \underline{W_{pred} W_{base}} \cdot \frac{\delta_S}{\sigma_X}, \quad \tilde{Y}_{norm} = \text{vec}_{12}(W_{pred} W_{base} S_{norm} + \Delta S_{fut}), \tag{18}$$

where vec_{12} denotes the vectorization operator implemented on the first (N'_{seg}) and the second dimensions ($\mathscr{P}$). This phase causes a two-fold disruption: (1) The perturbation vector is **multiplied sequentially by two linear weight** layers. (2) When the corrupted patches are unfolded back into the continuous output, the high-frequency phase shifts break the smooth temporal transitions between

adjacent lookback segments, which may cause **sharp jump discontinuities at patch boundaries**. Attack chains on other target PE TSF models are provided in Appendix A.3.

3.5 Main Theoretical Results

STDSA has two advantages with theoretical guarantees: **1. Disruption Lower and Upper Bounds:** The lower bound explains why PE TSF models cannot eliminate the attack perturbation, while the upper bound indicates the strongest disruption that can be caused by STDSA. **2. Defense Invalidation:** It explains why filtering defenses fail to remove the attack.

Theorem 1 (Disruption Lower and Upper Bounds)**.** *Let $X \in \mathscr{X}$ where the input sample domain $\mathscr{X}$ is a linear subspace of $\mathbb{R}^{L\times C}$, and $\hat{Y} = \mathcal{T}_{tgt}\tilde{X} \in \mathbb{R}^{H\times C}$ where $\mathcal{T}_{tgt}$ is the linear prediction operator of the target PE TSF model. It is reasonable to assume that $\mathcal{T}_{tgt}^{-1}(\{0\}) \cap \mathscr{X} = \{0\}$, otherwise $\mathcal{T}_{tgt}$ can map infinite samples in $\mathscr{X}$ to only one single prediction in $\mathbb{R}^{H\times C}$, which means that the representation of $\mathcal{T}_{tgt}$ partially collapses in $\mathscr{X}$. Then the Frobenius norm of the forecasting error perturbation $\|\Delta Y\|_F = \|\mathcal{T}_{tgt}\tilde{X} - \mathcal{T}_{tgt}X\|_F = \|\mathcal{T}_{tgt}\delta\|_F$ satisfies the following lower and upper bounds:*

$$0 < \sigma_{\min}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) \cdot \|\delta\|_F \leqslant \|\Delta Y\|_F \leqslant \sigma_{max}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) \cdot \sqrt{L\cdot C} \cdot \|\delta\|_\infty, \quad \forall \delta \in \mathscr{X}, \delta \neq 0. \tag{19}$$

where $\sigma_{min}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) > 0$ and $\sigma_{max}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) > 0$ denote the minimum and the maximum singular values of $\mathcal{T}_{tgt}$ restricted to $\mathscr{X}$, respectively.

The proof is given in Appendix A.1. This theorem indicates that the minimal disruption caused by STDSA is guaranteed by the minimum singular value of the target model. Moreover, the maximal disruption can be **further amplified** to $\sqrt{L\cdot C}$ times the perturbation budget $\epsilon = \|\delta\|_\infty$. If the lookback window and channels keep growing, the maximal disruption can still increase.

Theorem 2 (Defense Invalidation)**.** *Without loss of generality, let $\mathcal{T}_{dfs}\tilde{X} := \mathcal{F}_L^{-1}(W_{dfs} \odot \mathcal{F}_L\{\tilde{X}\})$ be a frequency-filtering defense mechanism with LPF matrix W_{dfs} and frequency cutoff K_{dfs}. Let $K_{min} = \min\{K_{dfs}, K_{keep}\}$ where K_{keep} denotes the frequency cutoff of STDSA (see Eq. 11). Then*

$$\mathcal{T}_{dfs}(X+\delta) = \mathcal{T}_{dfs}X + \mathcal{P}_{min}(\delta), \tag{20}$$

where $\mathcal{P}_{min}$ denotes the projection operator onto the subspace of $\mathscr{X}$ constructed by the frequency domain of cutoff K_{min}. Since $\mathcal{T}_{dfs}$ is a defense method that usually has to use a larger cutoff $K_{dfs} \geqslant K_{keep}$ to retain higher fidelity, $\mathcal{P}_{min}(\delta) = \delta$ in this case and the perturbation δ persists.

The proof is given in Appendix A.2.

4 Experiments

Our experimental setup strictly follows standard evaluation protocols in the literature of PE TSF models, where MSE is the main evaluating metric. Moreover, we report the MSE, relative MSE increase (RMI), and adversarial MSE reduction (AMR) results for the clean cases, the attacked cases, and the attacked plus defended cases for all the five target models and six defense measures. Details for these target models and defense measures are provided in Section 3.4, Appendices A.3 and A.4. We adopt three widely-used multivariate time series forecasting datasets: ETTh1 (Zhou et al., 2021), Weather[1] (Jena Climate Dataset), and Exchange (Lai et al., 2018), and an attack baseline: VMI-FGSM (Wang & He, 2021) plus our Two Structural Pathways (TSP) loss in (5). Since the original surrogates Inception (Szegedy et al., 2016; 2017) and Resnet (He et al., 2016) are not applicable to our TSF task, we use our TSP loss in (5) as the surrogate of VMI-FGSM. Detailed experimental settings are given in Appendix A.5. The main experiments are presented in the main text, while **additional experiments are provided in Appendix A.6, including comparisons with VMI-FGSM+Our TSP Loss and random perturbations, ablation study, and hyperparameter sensitivity analysis**.

4.1 Main Results on ETTh1, Weather, and Exchange

The attack and defense results on ETTh1, Weather, and Exchange are provided in Tables A1, 1, and A2. STDSA achieves strong disruption across all target models on all datasets by consistently inducing substantial disruption (higher MSE and RMI) while resisting the defense measures. Specifically,

[1] https://www.bgc-jena.mpg.de/wetter/

Table 1: (Defended)-clean → undefended-adversarial → defended-adversarial MSE trajectories on Weather under the STDSA attack. In the attack side, higher MSE (↑), higher RMI (↑), and lower AMR (↓) indicate better performance.

Defense	Attack Scheme	TimeBase	iTransformer	SparseTSF	FITS	LSINet	Model Macro Average
None	**STDSA**	MSE 0.1450 → 0.1804 → **0.1804** RMI **24.41**% → AMR 0.00%	MSE 0.1720 → 0.2189 → **0.2189** RMI **27.27**% → AMR 0.00%	MSE 0.1715 → 0.2261 → **0.2261** RMI **31.84**% → AMR 0.00%	MSE 0.1688 → 0.2241 → **0.2241** RMI **32.76**% → AMR 0.00%	MSE 0.1521 → 0.2037 → **0.2037** RMI **33.93**% → AMR 0.00%	MSE 0.1619 → 0.2106 → **0.2106** RMI **30.08**% → AMR 0.00%
	VMI-FGSM+Our TSP Loss	MSE 0.1450 → 0.1752 → 0.1752 RMI 20.83% → AMR 0.00%	MSE 0.1720 → 0.2091 → 0.2091 RMI 21.57% → AMR 0.00%	MSE 0.1715 → 0.2142 → 0.2142 RMI 24.90% → AMR 0.00%	MSE 0.1688 → 0.2110 → 0.2110 RMI 25.00% → AMR 0.00%	MSE 0.1521 → 0.1922 → 0.1922 RMI 26.36% → AMR 0.00%	MSE 0.1619 → 0.2003 → 0.2003 RMI 23.72% → AMR 0.00%
TV	**STDSA**	MSE 0.1450 → 0.1804 → **0.1803** RMI **24.41**% → AMR 0.06%	MSE 0.1720 → 0.2189 → **0.2191** RMI **27.27**% → AMR -0.09%	MSE 0.1715 → 0.2261 → **0.2261** RMI **31.84**% → AMR 0.00%	MSE 0.1689 → 0.2241 → **0.2240** RMI **32.68**% → AMR 0.04%	MSE 0.1524 → 0.2037 → **0.2043** RMI **33.66**% → AMR -0.29%	MSE 0.1620 → 0.2106 → **0.2107** RMI **30.00**% → AMR -0.05%
	VMI-FGSM+Our TSP Loss	MSE 0.1450 → 0.1752 → 0.1751 RMI 20.83% → AMR 0.06%	MSE 0.1720 → 0.2091 → 0.2096 RMI 21.57% → AMR -0.24%	MSE 0.1715 → 0.2142 → 0.2143 RMI 24.90% → AMR -0.05%	MSE 0.1689 → 0.2110 → 0.2109 RMI 24.93% → AMR 0.05%	MSE 0.1524 → 0.1922 → 0.1927 RMI 26.12% → AMR -0.26%	MSE 0.1620 → 0.2003 → 0.2005 RMI 23.64% → AMR -0.10%
Randomized Smoothing	**STDSA**	MSE 0.1444 → 0.1804 → **0.1792** RMI **24.93**% → AMR 0.67%	MSE 0.1721 → 0.2189 → **0.2191** RMI **27.19**% → AMR -0.09%	MSE 0.1714 → 0.2261 → **0.2259** RMI **31.91**% → AMR 0.09%	MSE 0.1688 → 0.2241 → **0.2231** RMI **32.76**% → AMR 0.45%	MSE 0.1526 → 0.2037 → **0.2050** RMI **33.49**% → AMR -0.64%	MSE 0.1618 → 0.2106 → **0.2105** RMI **30.16**% → AMR 0.05%
	VMI-FGSM+Our TSP Loss	MSE 0.1444 → 0.1752 → 0.1741 RMI 21.33% → AMR 0.63%	MSE 0.1721 → 0.2091 → 0.2098 RMI 21.50% → AMR -0.33%	MSE 0.1714 → 0.2142 → 0.2141 RMI 24.97% → AMR 0.05%	MSE 0.1688 → 0.2110 → 0.2104 RMI 25.00% → AMR 0.28%	MSE 0.1526 → 0.1922 → 0.1935 RMI 25.95% → AMR -0.68%	MSE 0.1618 → 0.2003 → 0.2004 RMI 23.79% → AMR -0.05%
AT	**STDSA**	MSE 0.1492 → 0.1804 → **0.1839** RMI **20.91**% → AMR -1.94%	MSE 0.1576 → 0.2189 → **0.2033** RMI **38.90**% → AMR 7.13%	MSE 0.1772 → 0.2261 → **0.2234** RMI **27.60**% → AMR 1.19%	MSE 0.1752 → 0.2241 → **0.2224** RMI **27.91**% → AMR 0.76%	MSE 0.1502 → 0.2037 → **0.1961** RMI **35.62**% → AMR 3.73%	MSE 0.1619 → 0.2106 → **0.2058** RMI **30.08**% → AMR 2.28%
	VMI-FGSM+Our TSP Loss	MSE 0.1492 → 0.1752 → 0.1763 RMI 17.43% → AMR -0.63%	MSE 0.1576 → 0.2091 → 0.1916 RMI 32.68% → AMR 8.37%	MSE 0.1772 → 0.2142 → 0.2135 RMI 20.88% → AMR 0.33%	MSE 0.1752 → 0.2110 → 0.2122 RMI 20.43% → AMR -0.57%	MSE 0.1502 → 0.1922 → 0.1855 RMI 27.96% → AMR 3.49%	MSE 0.1619 → 0.2003 → 0.1958 RMI 23.72% → AMR 2.25%
MSE Consistency	**STDSA**	MSE 0.1465 → 0.1804 → **0.1826** RMI **23.14**% → AMR -1.22%	MSE 0.1582 → 0.2189 → **0.2068** RMI **38.37**% → AMR 5.53%	MSE 0.1718 → 0.2261 → **0.2228** RMI **31.61**% → AMR 1.46%	MSE 0.1704 → 0.2241 → **0.2219** RMI **31.51**% → AMR 0.98%	MSE 0.1477 → 0.2037 → **0.1962** RMI **37.91**% → AMR 3.68%	MSE 0.1589 → 0.2106 → **0.2061** RMI **32.54**% → AMR 2.14%
	VMI-FGSM+Our TSP Loss	MSE 0.1465 → 0.1752 → 0.1748 RMI 19.59% → AMR 0.23%	MSE 0.1582 → 0.2091 → 0.1948 RMI 32.17% → AMR 6.84%	MSE 0.1718 → 0.2142 → 0.2116 RMI 24.68% → AMR 1.21%	MSE 0.1704 → 0.2110 → 0.2104 RMI 23.83% → AMR 0.28%	MSE 0.1477 → 0.1922 → 0.1848 RMI 30.13% → AMR 3.85%	MSE 0.1589 → 0.2003 → 0.1953 RMI 26.05% → AMR 2.50%
KL Temporal Consistency	**STDSA**	MSE 0.1480 → 0.1804 → **0.1843** RMI **21.89**% → AMR -2.16%	MSE 0.1723 → 0.2189 → **0.2234** RMI **27.05**% → AMR -2.06%	MSE 0.1699 → 0.2261 → **0.2240** RMI **33.08**% → AMR 0.93%	MSE 0.1686 → 0.2241 → **0.2237** RMI **32.92**% → AMR 0.18%	MSE 0.1485 → 0.2037 → **0.1977** RMI **37.17**% → AMR 2.95%	MSE 0.1614 → 0.2106 → **0.2106** RMI **30.48**% → AMR 0.00%
	VMI-FGSM+Our TSP Loss	MSE 0.1480 → 0.1752 → 0.1766 RMI 18.38% → AMR -0.80%	MSE 0.1723 → 0.2091 → 0.2107 RMI 21.36% → AMR -0.77%	MSE 0.1699 → 0.2142 → 0.2116 RMI 26.07% → AMR 1.21%	MSE 0.1686 → 0.2110 → 0.2107 RMI 25.15% → AMR 0.14%	MSE 0.1485 → 0.1922 → 0.1859 RMI 29.43% → AMR 3.28%	MSE 0.1614 → 0.2003 → 0.1991 RMI 24.10% → AMR 0.60%
AT + MSE Consistency + TV	**STDSA**	MSE 0.1496 → 0.1804 → **0.1844** RMI **20.59**% → AMR -2.22%	MSE 0.1542 → 0.2189 → **0.1964** RMI **41.96**% → AMR 10.28%	MSE 0.1772 → 0.2261 → **0.2231** RMI **27.60**% → AMR 1.33%	MSE 0.1753 → 0.2241 → **0.2220** RMI **27.84**% → AMR 0.94%	MSE 0.1518 → 0.2037 → **0.1966** RMI **34.19**% → AMR 3.49%	MSE 0.1616 → 0.2106 → **0.2045** RMI **30.32**% → AMR 2.90%
	VMI-FGSM+Our TSP Loss	MSE 0.1496 → 0.1752 → 0.1768 RMI 17.11% → AMR -0.91%	MSE 0.1542 → 0.2091 → 0.1864 RMI 35.60% → AMR 10.86%	MSE 0.1772 → 0.2142 → 0.2134 RMI 20.88% → AMR 0.37%	MSE 0.1753 → 0.2110 → 0.2121 RMI 20.37% → AMR -0.52%	MSE 0.1518 → 0.1922 → 0.1861 RMI 26.61% → AMR 3.17%	MSE 0.1616 → 0.2003 → 0.1950 RMI 23.95% → AMR 2.65%

Exchange shows the highest overall attack vulnerability, where STDSA drives the model macro average MSE from 0.1012 to 0.1633 and achieves a macro average RMI of 61.36%. Weather also shows strong attack vulnerability, where STDSA raises the macro average MSE from 0.1619 to 0.2106, which corresponds to a 30.08% macro average RMI. The degradation ranges from 24.41% on TimeBase to 33.93% on LSINet. Last, ETTh1 also experiences consistent disruption, where STDSA raises the macro average MSE from 0.3885 to 0.4549 for a 17.09% macro average RMI.

As for model-specific attack susceptibility, FITS and LSINet suffer the most severe performance drops under attack. FITS reaches the highest vulnerability on Exchange with a 76.17% RMI (MSE rises from 0.0873 to 0.1538), while LSINet shows high vulnerability on ETTh1 (19.42% RMI) and Weather (33.93% RMI). SparseTSF is also highly vulnerable across all the datasets, which yields 71.16% RMI on Exchange, 31.84% RMI on Weather, and 16.99% RMI on ETTh1. iTransformer and TimeBase show relatively higher resilience compared to lightweight architectures, but STDSA still degrades iTransformer by 39.91% RMI on Exchange and 15.51% RMI on ETTh1.

STDSA persists against the defense measures. Standard input-preprocessing defenses fail to remove the attack. TV yields near-zero AMRs (macro averages of 0.20% on ETTh1, -0.05% on Weather, and 0.24% on Exchange). Randomized smoothing also fails to recover clean time series and keeps macro average AMRs below 1.10% across all datasets. Even under advanced defenses like AT and KL, STDSA retains significant effectiveness. For example, on TimeBase and Weather dataset, the combined defense (AT + MSE consistency + TV) yields a negative AMR (-2.22%), which indicates that prediction error increases beyond the undefended attack baseline. Besides, STDSA achieves 100% **impact retention** with macro average AMR of 0.00% against KL on Weather.

Averaged over all three datasets and five target models, STDSA produces an RMI of 36.18%. These results indicate that the adversarial perturbations generated from the two structural pathways (direct-periodic) successfully disrupt different target structures, including temporal basis models, inverted-attention models, frequency-domain models, and sparse interaction models. Meanwhile, under the ℓ_∞ perturbation budget of $\epsilon = 0.1$, the measured averaged SNRs are 20.27dB, 26.41dB, and 17.89dB on ETTh1, Exchange, and Weather, respectively, with an overall average SNR of 21.52dB. This high overall SNR highlights the stealthiness of STDSA and confirms that the generated adversarial noise remains largely imperceptible relative to the underlying clean signals. Comparisons between STDSA and VMI-FGSM+Our TSP Loss are provided in Appendix A.6.2.

5 Conclusion

In this work, we reveal a critical structural vulnerability in parameter-efficient (PE) time series forecasting (TSF) models, which challenges optimistic assumptions regarding architectural minimalism and extreme parameter sparsity. We develop the Spectral-Temporal Decoupled Sparse Attack (STDSA), a black-box and non-query attack framework that simultaneously disrupts localized morphological shapes and dominant periodic trajectories. It integrates a hybrid direct-periodic loss landscape with segment-wise spectral-temporal decoupling via Real Fast Fourier Transform (RFFT) to mask high-frequency variations and simulate natural distribution drifts. We establish formal lower and upper bounds on forecasting error perturbations, and analytically prove why standard frequency-filtering defenses fail to neutralize the attack vectors.

Extensive experiments on the ETTh1, Weather, and Exchange datasets show that STDSA successfully disrupts five state-of-the-art PE TSF models under a tight ℓ_∞ perturbation budget of $\epsilon = 0.1$, corresponding to an overall average SNR of 21.52dB. Moreover, STDSA appears more persistent than the baseline VMI-FGSM+Our TSP Loss against six representative defense mechanisms with minimal impact retention, including adversarial training, randomized smoothing, total variation regularization, Kullback-Leibler divergence regularization, MSE consistency regularization, and a combined defense (AT + MSE consistency + TV). Beyond revealing security risks, STDSA can serve as an automated stress-testing engine for critical infrastructure domains and act as a diagnostic tool to evaluate structural information loss during edge-device compression. Future research will focus on developing structurally robust PE forecasting methods that withstand advanced multi-domain adversarial manipulations.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 09–15 Jun 2019.

Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2196–2205. PMLR, 13–18 Jul 2020.

Srinath Dama, Kevin Course, and Prasanth B. Nair. Shifting time: Time-series forecasting with khatri-rao neural operators. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 12402–12435. PMLR, 13–19 Jul 2025.

Daizong Ding, Mi Zhang, Fuli Feng, Yuanmin Huang, Erling Jiang, and Min Yang. Black-box adversarial attack on time series classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):7358–7368, Jun. 2023.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

Micah Goldblum, Liam Fowl, Soheil Feizi, and Tom Goldstein. Adversarially robust distillation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3996–4003, Apr. 2020.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

Swaminathan Gurumurthy, Shaojie Bai, Zachary Manchester, and J. Zico Kolter. Joint inference and input optimization in equilibrium networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 16818–16832. Curran Associates, Inc., 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Qihe Huang, Zhengyang Zhou, Kuo Yang, Zhongchao Yi, Xu Wang, and Yang Wang. TimeBase: The power of minimalism in efficient long-term time series forecasting. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 26227–26246. PMLR, 13–19 Jul 2025.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-llm: Time series forecasting by reprogramming large language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 23857–23880, 2024.

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*, 2022.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pp. 95–104. Association for Computing Machinery, 2018.

Zhao-Rong Lai and Weiwen Wang. Invariant risk minimization is a total variation model. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 25913–25935. PMLR, 21–27 Jul 2024.

Zhao-Rong Lai, Xiwen Yuan, Ziliang Chen, Liangda Fang, and Yongsen Zheng. A unified total variation framework for membrane potential perturbation dynamic. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Zhao-Rong Lai, Xiwen Yuan, and Jian Weng. Adversarial attack and defense for denoising diffusion sampling. In *The 43rd International Conference on Machine Learning*, 2026b.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and Junjie Yang. SparseTSF: Modeling long-term time series forecasting with *1k* parameters. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30211–30226. PMLR, 21–27 Jul 2024.

Linbo Liu, Youngsuk Park, Trong Nghia Hoang, Hilaf Hasson, and Jun Huan. Robust multivariate time-series forecasting: Adversarial attacks and defense mechanisms. In *The Eleventh International Conference on Learning Representations*, 2023.

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 B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 11116–11140, 2024.

Aitian Ma, Dongsheng Luo, and Mo Sha. Mixlinear: Extreme low resource multivariate time series forecasting with $0.1k$ parameters. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiaowen Ma, Zhen-Liang Ni, Shuai Xiao, and Xinghao Chen. TimePro: Efficient multivariate long-term time series forecasting with variable- and time-aware hyper-state. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 42096–42111. PMLR, 13–19 Jul 2025.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

Abrar Majeedi, Viswanatha Reddy Gajjala, Satya Sai Srinath Namburi Gnvv, Nada Magdi Elkordi, and Yin Li. LETS forecast: Learning embedology for time series forecasting. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 42668–42690. PMLR, 13–19 Jul 2025.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25, pp. 1185–1196, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712456.

Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1924–1933, June 2021.

Yuanchao Wang, Zhao-Rong Lai, and Tianqi Zhong. Out-of-distribution generalization for total variation based invariant risk minimization. In *The Thirteenth International Conference on Learning Representations*, 2025.

Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Zhijian Xu, Ailing Zeng, and Qiang Xu. Fits: Modeling time series with 10k parameters. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 26295–26318, 2024.

Hongzhan Yu and Sicun Gao. Activation-descent regularization for input optimization of relu networks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57441–57458. PMLR, 2024.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7472–7482. PMLR, 09–15 Jun 2019.

Xu Zhang, Qitong Wang, Peng Wang, and Wei Wang. A lightweight sparse interaction network for time series forecasting. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press, 2025a. ISBN 978-1-57735-897-8.

Xu Zhang, Kaidi Xu, Ziqing Hu, and Ren Wang. Optimizing robustness and accuracy in mixture of experts: A dual-model approach. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 76436–76450. PMLR, 13–19 Jul 2025b.

Lifan Zhao and Yanyan Shen. Proactive model adaptation against concept drift for online time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25, pp. 2020–2031, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712456.

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.

Tian Zhou, Peisong Niu, xue wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained lm. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 43322–43355. Curran Associates, Inc., 2023.

A Appendix

A.1 Proof of Theorem 1

Proof. The condition $\mathcal{T}_{tgt}^{-1}(\{0\}) \cap \mathscr{X} = \{0\}$ indicates that no non-zero vector in the input subspace $\mathscr{X}$ maps to zero under $\mathcal{T}_{tgt}$. As a result, the restricted linear operator $\mathcal{T}_{tgt}\big|_{\mathscr{X}}$ is injective, and $\sigma_{min}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) > 0$. Since the perturbation δ satisfies (14), we have $(X+\delta) \in \mathscr{X}$. Since $X \in \mathscr{X}$ and $\mathscr{X}$ is a linear subspace of $\mathbb{R}^{H\times C}$, we have $\delta := (X+\delta) - X \in \mathscr{X}$.

By the definition of the minimum singular value of a linear operator acting on a vector space:

$$\sigma_{\min}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) = \min_{\delta\in\mathscr{X},\delta\neq 0} \frac{\|\mathcal{T}_{tgt}\delta\|_F}{\|\delta\|_F} > 0. \tag{21}$$

Rearranging this inequality yields:

$$\|\mathcal{T}_{tgt}\delta\|_F \geqslant \sigma_{\min}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) \cdot \|\delta\|_F > 0, \quad \forall \delta \in \mathscr{X}, \delta \neq 0. \tag{22}$$

On the other hand, $\|\delta\|_F$ can be bounded by $\|\delta\|_\infty$ as follows:

$$\|\delta\|_F = \sqrt{\sum_{i=1}^{L}\sum_{j=1}^{C}\delta_{i,j}^2} \leqslant \sqrt{\sum_{i=1}^{L}\sum_{j=1}^{C}(\max_{k,l}(\delta_{k,l})^2)} = \sqrt{L\cdot C\cdot\|\delta\|_\infty^2} = \sqrt{L\cdot C}\cdot\|\delta\|_\infty. \tag{23}$$

Then the upper bound can be deduced as follows:

$$\|\mathcal{T}_{tgt}\delta\|_F \leqslant \sigma_{\max}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) \cdot \|\delta\|_F \leqslant \sigma_{max}(\mathcal{T}_{tgt}\big|_{\mathscr{X}}) \cdot \sqrt{L\cdot C}\cdot\|\delta\|_\infty, \quad \forall \delta \in \mathscr{X}, \delta \neq 0. \tag{24}$$

□

A.2 Proof of Theorem 2

Proof. The adversarial perturbation $\delta \in \mathbb{R}^{L\times C}$ of STDSA is constructed via IRFFT of low-frequency bins $\omega \leqslant K_{keep}$ in (10):

$$\mathcal{F}_{RFFT}\{\delta\}(\omega) = 0, \quad \forall \omega > K_{keep}. \tag{25}$$

Applying the defense filter W_{dfs} with cutoff K_{dfs} on δ yields:

$$\begin{aligned}
&\left(W_{\text{dfs}} \odot \mathcal{F}_L\{\delta\}\right)(\omega)\\
=&(W_{dfs} \odot \mathcal{F}_L|_{\mathbb{R}^{L\times C}}\{\delta\})(\omega)\\
=&(W_{dfs} \odot \mathcal{F}_{RFFT}\{\delta\})(\omega)\\
=&\begin{cases}\mathcal{F}_{RFFT}\{\delta\}(\omega), & \text{if } \omega \leqslant K_{min}\\ 0, & \text{otherwise}\end{cases}.
\end{aligned} \tag{26}$$

Since $\delta \in \mathbb{R}^{L\times C}$, the complex-valued DFT operator $\mathcal{F}_L$ is equivalent to its restriction on $\mathbb{R}^{L\times C}$, which is $\mathcal{F}_{RFFT}$. Moreover, considering both frequency cutoffs from STDSA and the defense, $\mathcal{F}_{RFFT}\{\delta\}(\omega)$ will be nonzero only if $\omega \leqslant K_{min} = \min\{K_{dfs}, K_{keep}\}$.

Applying IDFT $\mathcal{F}_L^{-1}$ to both sides of (26) for $\omega \leqslant K_{min}$ yields:

$$\begin{aligned}
&\mathcal{F}_L^{-1}(W_{dfs} \odot \mathcal{F}_L\{\delta\})(\omega)\\
=&\mathcal{F}_L^{-1}|_{\mathbb{R}^{L\times C}}\mathcal{F}_{RFFT}\{\delta\}(\omega)\\
=&\mathcal{F}_{IRFFT}\mathcal{F}_{RFFT}\{\delta\}(\omega)\\
=&\delta(\omega), \qquad \forall \omega \leqslant K_{min}.
\end{aligned} \tag{27}$$

For $\omega > K_{min}$, (26) yields:

$$\mathcal{F}_L^{-1}(W_{dfs} \odot \mathcal{F}_L\{\delta\})(\omega) = 0, \ \forall \omega > K_{min}. \tag{28}$$

Combining (27) and (28), we have

$$\mathcal{T}_{dfs}\delta = \mathcal{F}_L^{-1}\left(W_{\text{dfs}} \odot \mathcal{F}_L\{\delta\}\right)$$

$$
\begin{aligned}
&= \sum_{\omega=1}^{K_{\min}} \delta(\omega) \\
&= \mathcal{P}_{min}(\delta).
\end{aligned} \tag{29}
$$

Moreover, as the defense filter has to retain a large proportion of the original signal in order to keep a high fidelity, we usually have $K_{dfs} \geqslant K_{keep}$. In this case, $\mathcal{P}_{min}(\delta) = \delta$ and the perturbation persists.

□

A.3 Attack Chains on Other Time Series Forecasting Models

A.3.1 Attack Chain on FITS

We have already introduced FITS in Section 2.1. The proposed STDSA exploits the reliance of FITS on low-frequency dominance by constraining the adversarial perturbation δ to a low-frequency subspace with frequency cutoff $K_{keep} \leqslant K_{cut}$. It exposes at least three vulnerabilities to STDSA.

Filter bypass identity. Because δ is generated via IRFFT restricted to frequency bins $\omega \leqslant K_{keep} \leqslant K_{cut}$, the spectrum of δ contains zero energy above K_{cut}. From Theorem 2, W_{LPF} acts as an exact identity operator on δ:

$$
\begin{aligned}
W_{LPF} \odot \mathcal{F}_L\{\delta\} &= W_{LPF} \odot \mathcal{F}_L|_{\mathbb{R}^{L\times C}}\{\delta\} \\
&= W_{LPF} \odot \mathcal{F}_{RFFT}\{\delta\} \\
&= \mathcal{F}_{RFFT}\{\delta\}.
\end{aligned} \tag{30}
$$

Thus the core architectural defense of FITS, which is the high-frequency LPF truncation, fails to eliminate the adversarial perturbation δ.

Perturbation linear propagation. The perturbation propagates directly into the complex linear layer W_f and scales linearly:

$$
\Delta Y = \mathcal{F}_H^{-1}\left(\mathcal{P}_H\left(W_f\left(W_{LPF} \odot \mathcal{F}_L\{\delta\}\right)\right)\right) = \mathcal{F}_H^{-1}\mathcal{P}_H W_f \mathcal{F}_{RFFT}\{\delta\}. \tag{31}
$$

Disruption error bounds. We define $\mathcal{T}_{FITS}$ as the **end-to-end linear operator** that maps input perturbation δ to prediction error ΔY. Then $\mathcal{T}_{FITS}$ **satisfies** the disruption error bounds in Theorem 1:

$$
0 < \sigma_{\min}(\mathcal{T}_{FITS}\big|_{\mathscr{X}}) \cdot \|\delta\|_F \leqslant \|\Delta Y\|_F \leqslant \sigma_{max}(\mathcal{T}_{FITS}\big|_{\mathscr{X}}) \cdot \sqrt{L \cdot C} \cdot \|\delta\|_\infty, \quad \forall \delta \in \mathscr{X}. \tag{32}
$$

A.3.2 Attack Chain on iTransformer

iTransformer inverts the standard time-series Transformer architecture by taking the entire lookback sequence of a perturbed individual variable $\tilde{X}_{:,c} \in \mathbb{R}^L$ as a single temporal token. Instead of calculating self-attention across time steps, iTransformer embeds time series into high-dimensional representations with a linear layer and applies multi-head self-attention across different channels (variates), followed by channel-independent linear projections to forecast the future horizon H.

Normalization distortion. iTransformer normalizes each variable independently using RevIN-style statistics:

$$
\mu_c(\tilde{X}) = \frac{1}{L}\mathbb{1}_L^\top \tilde{X}_{:,c}, \quad \sigma_c(\tilde{X}) = \sqrt{\frac{1}{L}\|\tilde{X}_{:,c} - \mu_c(\tilde{X})\mathbb{1}_L\|_2^2 + \varepsilon_{var}}, \tag{33}
$$

where $\tilde{X}_{:,c}$ denotes the c-th channel of $\tilde{X}$, and ε_{var} is a small positive numerical stability constant set to 10^{-5} by default. Under perturbation $\delta_{:,c}$, the perturbed normalized sequence $\tilde{X}_{\text{norm},c}$ can be formulated as:

$$
\tilde{X}_{norm,c} = \frac{(X_{:,c} + \delta_{:,c}) - \mu_c(\tilde{X})\mathbb{1}_L}{\sigma_c(\tilde{X})} = X_{norm,c} \cdot \frac{\sigma_c(X)}{\sigma_c(\tilde{X})} + \delta_{norm,c}, \delta_{norm,c} = \frac{(\mathbb{I}_L - \frac{1}{L}\mathbb{1}_L\mathbb{1}_L^\top)\delta_{:,c}}{\sigma_c(\tilde{X})}. \tag{34}
$$

Not only the clean input $X_{norm,c}$ will be disrupted by multiplying a factor $\frac{\sigma_c(X)}{\sigma_c(\tilde{X})}$, but also the perturbation $\delta_{norm,c}$ persists.

Cross-variate attention disruption. Each variable token $E_c \in \mathbb{R}^{\mathfrak{d}}$ is embedded via matrix $W_{emb} \in \mathbb{R}^{\mathfrak{d}\times L}$:

$$\tilde{E}_c = W_{emb}\tilde{X}_{norm,c} = E_c \cdot \frac{\sigma_c(X)}{\sigma_c(\tilde{X})} + \Delta E_c, \quad \Delta E_c = W_{emb}\delta_{norm,c}, \tag{35}$$

where $\mathfrak{d}$ denotes the total model dimensionality. Because STDSA optimizes δ within low-frequency components $\omega < K_{keep}$, the vector $\delta_{norm,c}$ lies entirely in the smooth range of W_{emb}, which prevents energy dissipation.

Let $E = [E_1, \ldots, E_C]^\top \in \mathbb{R}^{C\times\mathfrak{d}}$ be the full token matrix. Under perturbation ΔE, the Query, Key, and Value matrices are projected as:

$$\tilde{Q} = (E + \Delta E)W^Q, \quad \tilde{K} = (E + \Delta E)W^K, \quad \tilde{V} = (E + \Delta E)W^V. \tag{36}$$

Then the attention weights $\tilde{A} = \mathrm{softmax}\left(\frac{\tilde{Q}\tilde{K}^\top}{\sqrt{\mathfrak{d}_m}}\right) \in \mathbb{R}^{C\times C}$ experience a shift ΔA:

$$\begin{aligned}\Delta A &= \tilde{A} \odot \left(\Delta\mathscr{B} - \mathrm{rowsums}(\tilde{A} \odot \Delta\mathscr{B})\mathbb{1}^\top\right), \\ \Delta\mathscr{B} &:= \frac{1}{\sqrt{\mathfrak{d}_m}}\left(\Delta E W^Q \tilde{K}^\top + \tilde{Q}(W^K)^\top(\Delta E)^\top\right),\end{aligned} \tag{37}$$

where $\odot$ and $\mathrm{rowsums}$ denote the Hadamard product and the rowsum operator, respectively, $\mathfrak{d}_m$ denotes the per-head projection dimensionality. The corresponding encoder output $\tilde{\mathscr{E}} \in \mathbb{R}^{C\times\mathfrak{d}}$ becomes (first-order Taylor approximation):

$$\tilde{\mathscr{E}} = \tilde{A}\tilde{V} = (A + \Delta A)(V + \Delta V) \approx \mathscr{E} + \underbrace{A\Delta E W^V}_{\text{direct error}} + \underbrace{\Delta A V}_{\text{cross-variate error}}. \tag{38}$$

The term ΔAV indicates that an adversarial perturbation designed for a single variable propagates into all other variable representations.

Linear projection and de-normalization compounding. The linear projection layer $W_{\mathrm{proj}} \in \mathbb{R}^{H\times\mathfrak{d}}$ maps representations to the output horizon:

$$\tilde{Y}_{norm,c} = W_{proj}\tilde{\mathscr{E}}_c = Y_{norm,c} + W_{proj}\Delta\mathscr{E}_c. \tag{39}$$

Restoring scale via instance de-normalization yields the total output forecast error $\Delta Y_c = \tilde{Y}_c - Y_c$:

$$\Delta Y_c \approx \sigma_c(\tilde{X})W_{proj}\Delta\mathscr{E}_c + \frac{1}{L}\mathbb{1}_H\mathbb{1}_L^\top\delta_{:,c} + \mathcal{O}(\|\delta\|_2^2). \tag{40}$$

This disruption comes from both the perturbed tokens $\mathscr{E}_c$ and the perturbation δ itself.

A.3.3 Attack Chain on SparseTSF

SparseTSF is an ultra-lightweight long-term TSF model designed to achieve PE by combining mean normalization, one-dimensional convolutional aggregation, periodic downsampling, and cross-segment linear forecasting. The linearity and periodic segment downsampling make SparseTSF vulnerable to STDSA.

Convolutional linear propagation and low-pass bypassing. Since SparseTSF employs a channel-wise scheme, we omit the channel index c for simplicity. It uses centralization as the normalization scheme:

$$\tilde{X}_{norm} = X_{norm} + \delta_{norm}, \quad \delta_{norm} = (\mathbb{I}_L - \frac{1}{L}\mathbb{1}_L\mathbb{1}_L^\top)\delta. \tag{41}$$

Let $\mathcal{C}$ be the linear one-dimensional convolution. The aggregated sequence yields:

$$\begin{aligned}\tilde{X}_{conv} =& \mathcal{C}(X_{norm} + \delta_{norm}) + \underbrace{(X_{norm} + \delta_{norm})}_{\text{Residual Connection}} \\ =& (\mathcal{C}X_{norm} + X_{norm}) + (\mathcal{C}\delta_{norm} + \delta_{norm}) \\ =& X_{conv} + \delta_{conv}.\end{aligned} \tag{42}$$

Because STDSA constructs δ within low-frequency cross-segment bins ($\omega < K_{keep}$), the local kernel in one-dimensional convolution cannot smooth out δ_{norm}. Moreover, the residual connection also directly passes the entire δ_{norm} to $\tilde{X}_{conv}$.

Folding and linear forecasting projection. Using a similar folding scheme to (16), we have:

$$\tilde{S} = \mathcal{P}_{\mathscr{P}}(X_{conv} + \delta_{conv}) = \mathcal{P}_{\mathscr{P}} X_{conv} + \mathcal{P}_{\mathscr{P}} \delta_{conv} = S + \delta_S, \tag{43}$$

where $\mathcal{P}_{\mathscr{P}}$ denotes the folding operator in (16), which is a linear projection. Applying the segment forecasting layer W_{fcs} yields:

$$\tilde{Y}_{seg} = (S + \delta_S) W_{fcs}^{\top} = S W_{fcs}^{\top} + \delta_S W_{fcs}^{\top} = Y_{seg} + \delta_Y, \tag{44}$$

where δ_Y is the forecast perturbation.

Final unfolded output error and singular value bounds. Unfolding $\tilde{Y}_{seg}$ and restoring mean statistics produces output forecast deviation $\Delta Y = \tilde{Y} - Y$:

$$\Delta Y = \mathcal{P}_{\mathscr{P}}^{-1}(\delta_S W_{fcs}^{\top}) + \frac{1}{L} \mathbb{1}_H \mathbb{1}_L^{\top} \delta, \tag{45}$$

where $\mathcal{P}_{\mathscr{P}}^{-1}$ denotes the unfolding operator corresponding to $\mathcal{P}_{\mathscr{P}}$. This disruption comes from both forecasting and the residual connection.

A.3.4 Attack Chain on LSINet

LSINet is designed to capture spatial-temporal correlations across time patches and variables. It employs learned patch memory embeddings and Gumbel-Softmax discrete gating to construct sparse patch-interaction topology matrices.

Input normalization and patch propagation. LSINet uses the same RevIN-normalization as (34). After transforming into N_{seg} patches of length $\mathscr{P}$, the linear patch projection $W^{\mathscr{P}} \in \mathbb{R}^{\mathfrak{d} \times \mathscr{P}}$ transforms the patched features z_{ptc}:

$$\tilde{z} = W^{\mathscr{P}} z_{ptc} + W^{\mathscr{P}} \delta_{ptc} = z \cdot \left(\frac{\sigma_c}{\tilde{\sigma}_c} \right) + \delta_z, \quad \delta_z = W^{\mathscr{P}} \delta_{ptc}. \tag{46}$$

Because STDSA constructs δ by zeroing out high-frequency bins ($\omega \geqslant K_{keep}$), δ_{ptc} contains little high-frequency noise that would otherwise be relieved by linear smoothing. Hence $W^{\mathscr{P}}$ preserves the full energy of δ_z.

Sparse connection matrix bypassing. LSINet computes patch interactions via learned patch embeddings $E \in \mathbb{R}^{N_{seg} \times \mathfrak{d}}$ via Gumbel-Softmax:

$$E_{rec} = W_{cnct} E_{send}, \quad A_{sps} = \mathscr{G}(E_{rec}) \in \{0, 1\}^{N_{heads} \times N_{seg} \times N_{seg}}, \tag{47}$$

where $\mathscr{G}$ denotes the Gumbel-Softmax function, W_{cnct} denotes the connection learning matrix for the sparse structure, E_{send} denotes the sender memory embeddings, E_{rec} denotes the receiver memory embeddings, and N_{heads} denotes the number of attention heads, respectively. Since E depends solely on patch position indices rather than dynamic sequence values, the learned interaction topology matrix A_{sps} remains largely unchanged under input perturbation δ. The perturbed features propagate through the value projection W^V:

$$\tilde{z}_{out} = \mathcal{T}_{TU}\left(A_{sps} \cdot W^V(\tilde{z}) + \tilde{z} \right) = z_{out} + \Delta z_{out}, \tag{48}$$

where $\mathcal{T}_{TU}$ denotes the lightweight Multi-Layer Perceptron (MLP) that implements the time patch updating.

A.4 Six Defense Methods

A.4.1 Total Variation (TV) Regularization

TV regularization provides an input purification defense against adversarial perturbations in time series by enforcing temporal smoothness across consecutive time steps. Given a perturbed input

time series sequence $\tilde{X} \in \mathbb{R}^{B\times L\times C}$ with batch size B, lookback window length L, and C channels, the TV regularized objective solves the unconstrained minimization problem:

$$\min_{Z\in\mathbb{R}^{B\times L\times C}} \mathcal{L}_{TV}(Z;\tilde{X}) := \frac{1}{2}\|Z-\tilde{X}\|_F^2 + \lambda\cdot\mathcal{T}_{TV}(Z),$$
$$\mathcal{T}_{TV}(Z) := \sum_{b=1}^{B}\sum_{c=1}^{C}\sum_{t=1}^{L-1}|Z_{b,t+1,c}-Z_{b,t,c}|, \tag{49}$$

where λ denotes the regularization strength, and $\mathcal{T}_{TV}(Z)$ denotes the discrete isotropic TV functional computed along the temporal dimension L.

Because the absolute value function $|\cdot|$ is non-differentiable at zero, its subdifferential is given by the sign operator $\partial|\cdot| = \text{sign}(\cdot)$. We define the forward temporal finite difference as:

$$\Delta Z_{b,t,c} = Z_{b,t+1,c} - Z_{b,t,c}, \quad t\in\{1,\dots,L-1\}. \tag{50}$$

The element-wise subgradient matrix $\mathcal{S}\in\mathbb{R}^{B\times(L-1)\times C}$ is:

$$\mathcal{S}_{b,t,c} = \text{sign}(\Delta Z_{b,t,c}) \in \partial|\Delta Z_{b,t,c}|, \tag{51}$$

where $\partial|\Delta Z_{b,t,c}|$ denotes the subgradient set of $|\Delta Z_{b,t,c}|$. The full subgradient of the TV regularizer $[\nabla\mathcal{T}_{TV}(Z)]\in\mathbb{R}^{B\times L\times C}$ w.r.t. $Z_{b,t,c}$ is:

$$[\nabla\mathcal{T}_{TV}(z)]_{b,t,c} = \begin{cases} -\mathcal{S}_{b,1,c}, & t=1 \\ \mathcal{S}_{b,t-1,c} - \mathcal{S}_{b,t,c}, & 1<t<L \\ \mathcal{S}_{b,L-1,c}, & t=L \end{cases}. \tag{52}$$

Using step size γ for O iterations, the recurrence update rule initialized at $Z^{(0)}=\tilde{X}$ is given by:

$$Z^{(o+1)} = Z^{(o)} - \gamma\left((Z^{(o)}-\tilde{X}) + \lambda\cdot\nabla\mathcal{T}_{TV}(Z^{(o)})\right), \quad o\in\{0,1,\dots,O-1\}. \tag{53}$$

Then the recovered time series sequence is $X_{rcv} = Z^{(O)}$.

A.4.2 Randomized Smoothing

Randomized smoothing converts a deterministic base forecasting model into a noise-smoothed estimator with guaranteed robustness bounds against adversarial perturbations. Given a target long-term time series forecasting model $f:\mathbb{R}^{L\times C}\to\mathbb{R}^{H\times C}$ and a perturbed input sequence $\tilde{X}\in\mathbb{R}^{L\times C}$, the randomized smoothed estimator $g:\mathbb{R}^{L\times C}\to\mathbb{R}^{H\times C}$ is defined as the expectation of f under isotropic Gaussian noise perturbations:

$$g(\tilde{X}) = \mathbb{E}_{\varsigma\sim\mathcal{N}(0,\sigma_{rs}^2\mathbb{I}_{L\times C})}\left[f(\tilde{X}+\varsigma)\right], \tag{54}$$

where $\sigma_{rs}>0$ represents the Gaussian noise standard deviation hyperparameter for the ramdomized smoothing, and $\mathbb{I}_{L\times C}$ denotes the identity covariance tensor over the vectorization of the input space $\mathbb{R}^{L\times C}$. Next, an empirical Monte Carlo (MC) estimator $\hat{g}(\tilde{X})$ using N_{rs} independent and identically distributed (i.i.d.) Gaussian noise samples is constructed:

$$\hat{g}(\tilde{X}) = \frac{1}{N_{rs}}\sum_{m=1}^{N_{rs}} f\left(\tilde{X}+\varsigma^{(m)}\right), \quad \text{where } \varsigma^{(m)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,\sigma_{rs}^2\mathbb{I}_{L\times C}). \tag{55}$$

A.4.3 Adversarial Training (AT)

AT minimizes the expected forecasting empirical risk under perturbations:

$$\min_{\theta}\mathcal{L}_{AT}(\theta) = \left[\frac{1}{N}\sum_{n=1}^{N}\mathcal{L}_{tgt}(f_\theta(\tilde{X}_n),Y_n)\right], \tag{56}$$

where $(\tilde{X}_n,Y_n)$ denotes the n-th perturbed sample, f_θ denotes the target forecasting model with parameter θ, and $\mathcal{L}_{tgt}$ denotes the target loss.

A.4.4 KULLBACK-LEIBLER (KL) DIVERGENCE REGULARIZATION

To adapt KL divergence regularization to temporal regression tasks, the temporal sequences are mapped into normalized probability distributions along the forecasting horizon H. We adopt the same notations in Section A.4.3 and denote $\hat{Y}_{cln} = f_\theta(X) \in \mathbb{R}^{H\times C}$ and $\hat{Y}_{adv} = f_\theta(\tilde{X}) \in \mathbb{R}^{H\times C}$ as the output predictions on the clean input and the perturbed input, respectively. To compute divergence over continuous temporal forecasts, a temperature-scaled softmax operator with temperature hyperparameter $\tau > 0$ converts the temporal predictions along H into categorical temporal probability distributions:

$$\mathfrak{q}_{cln}(t,c) = \frac{\exp\left(\frac{\hat{Y}_{cln}(t,c)}{\tau}\right)}{\sum_{h=1}^{H}\exp\left(\frac{\hat{Y}_{cln}(h,c)}{\tau}\right)}, \quad \mathfrak{q}_{adv}(t,c) = \frac{\exp\left(\frac{\hat{Y}_{adv}(t,c)}{\tau}\right)}{\sum_{h=1}^{H}\exp\left(\frac{\hat{Y}_{adv}(h,c)}{\tau}\right)}, t \in [H], c \in [C]. \tag{57}$$

The temporal KL consistency loss $\mathcal{L}_{KL}(\hat{Y}_{adv}, \hat{Y}_{cln})$ measures the distribution shift induced by the adversarial perturbation relative to the detached clean reference prediction $\mathrm{dt}(\hat{Y}_{cln})$:

$$\mathcal{L}_{KL}(\hat{Y}_{adv}, \mathrm{dt}(\hat{Y}_{cln})) = \tau^2 \cdot \frac{1}{C}\sum_{c=1}^{C}\sum_{t=1}^{H}\mathrm{dt}(\mathfrak{q}_{cln}(t,c))\log\left(\frac{\mathrm{dt}(\mathfrak{q}_{cln}(t,c))}{\mathfrak{q}_{adv}(t,c)}\right). \tag{58}$$

The overall optimization objective minimizes the weighted sum of the clean reconstruction target loss $\mathcal{L}_{tgt}$ and $\mathcal{L}_{KL}$ controlled by hyperparameter $\beta > 0$:

$$\min_\theta \mathcal{L}_{tot}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\left(\mathcal{L}_{tgt}(\hat{Y}_{cln,n}, Y_n) + \beta \cdot \mathcal{L}_{KL}(\hat{Y}_{adv,n}, \mathrm{dt}(\hat{Y}_{cln,n}))\right). \tag{59}$$

A.4.5 MEAN SQUARED ERROR (MSE) CONSISTENCY REGULARIZATION

The MSE consistency regularization balances forecasting accuracy and adversarial robustness by constraining prediction variance between clean and perturbed inputs:

$$\mathcal{L}_{css}(X, \tilde{X}) = \frac{1}{H\cdot C}\|f_\theta(\tilde{X}) - \mathrm{dt}(f_\theta(X))\|_F^2. \tag{60}$$

The total loss is the weighted sum of $\mathcal{L}_{tgt}$ and $\mathcal{L}_{css}$ controlled by hyperparameter $\iota > 0$:

$$\min_\theta \mathcal{L}_{tot}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\left(\mathcal{L}_{tgt}(f_\theta(X_n), Y_n) + \iota \cdot \mathcal{L}_{css}(X_n, \tilde{X}_n)\right). \tag{61}$$

A.4.6 COMBINED DEFENSE WITH AT, MSE CONSISTENCY, AND TV

We also integrate AT, MSE consistency regularization, and TV regularization into a combined defense:

$$\min_\theta \mathcal{L}_{tot}(\theta) = \frac{1}{N}\sum_{n=1}^{N}\left(\mathcal{L}_{AT}(X_n, \tilde{X}_n) + \iota \cdot \mathcal{L}_{css}(X_n, \tilde{X}_n) + \lambda \cdot \mathcal{T}_{TV}(\tilde{X}_n)\right). \tag{62}$$

A.5 EXPERIMENTAL SETTINGS

A.5.1 DATASETS AND TARGET MODELS

ETTh1. It contains 17420 hourly observations with 7 variables, including the oil temperature of an electricity transformer and six power load features.

Exchange. It consists of 7588 daily observations of 8 foreign exchange rates.

Weather. It contains 52696 observations of 21 meteorological variables sampled every ten minutes.

We follow the standard chronological training, validation, and test splits used in long-term TSF. The normalization statistics are estimated only with the training split. For all datasets, we perform multivariate forecasting with a lookback length of $L = 720$ and a prediction horizon of $H = 96$.

The five target models represent different PE forecasting mechanisms, including temporal basis decomposition, inverted attention, sparse periodic mapping, frequency interpolation, and sparse variable interaction. They are trained according to their default settings. For each dataset, all target models are evaluated by the same ordered clean test samples and the same adversarial samples.

A.5.2 Attack Configuration

The mixing parameter in the two-pathway adversarial loss (5) is set to $w_1 = 0.9$. The perturbation budget is set to $\epsilon = 0.1$ under the ℓ_∞ perturbation constraint, which corresponds to an averaged high SNR of about 20dB. Thus the attack strength is relatively low. The SNR is calculated by

$$SNR = 10 \log_{10} \frac{\|X\|_F^2}{\|\tilde{X} - X\|_F^2}, \tag{63}$$

where X and $\tilde{X}$ denote the clean and adversarial inputs, respectively. The attack batch size is set to 16, the number of attack iterations is set to $\mathscr{K} = 200$, and the step size is set to $\eta = 0.01$. The attack is initialized from the clean input without a random start. At every attack iteration, the spectral projection retains $K_{\text{keep}} = 12$ low-frequency components, and the spectral blending hyperparameter is set to $\alpha = 0.75$. The sensitivity analysis of the key hyperparameters is provided in Appendix A.8. For each dataset, STDSA generates only one adversarial test set, which is subsequently evaluated on all five target models. It does not need to access the target models.

We use MSE as the primary forecasting-error metric and additionally report mean absolute error (MAE), root mean squared error (RMSE), and relative squared error (RSE). Let MSE_{cln}, MSE_{adv}, and MSE_{dfs} be the forecasting errors on clean, adversarial, and defended inputs, respectively. We quantify the relative attack-induced degradation by the **relative MSE increase (RMI)**:

$$\text{RMI} = \frac{\text{MSE}_{adv} - \text{MSE}_{cln}}{\text{MSE}_{cln}} \times 100\%. \tag{64}$$

This is the primary metric that evaluates the **disruptive effect** of STDSA. Besides, we also use the following adversarial MSE reduction (AMR) score to evaluate the defense performance:

$$\text{AMR} = \frac{\text{MSE}_{adv} - \text{MSE}_{dfs}}{\text{MSE}_{adv}} \times 100\%. \tag{65}$$

All forecasting metrics are calculated globally after collecting the predictions and ground-truth sequences from all test batches. The perturbation budget is fixed across different datasets, while the SNR is measured from the generated adversarial inputs.

A.6 Additional Experiments

A.6.1 Experimental Results on ETTh1 (Table A1) and Exchange (Table A2)

A.6.2 Comparison with VMI-FGSM+Our TSP Loss

As shown in Tables 1, A1, and A2, STDSA consistently outperforms VMI-FGSM + Our TSP Loss across all three datasets and achieves effectiveness under both undefended and defended settings. In the undefended setting, STDSA generates significantly more destructive perturbations than VMI-FGSM + Our TSP Loss across all target models. On Weather, STDSA achieves a macro-average adversarial MSE of 0.2106 and an RMI of 30.08%, compared to 0.2003 MSE and 23.72% RMI for VMI-FGSM + Our TSP Loss. On ETTh1, STDSA elevates the macro-average adversarial MSE to 0.4549 (RMI 17.09%), whereas VMI-FGSM + Our TSP Loss reaches 0.4397 (RMI 13.18%). On Exchange, STDSA attains a macro-average adversarial MSE of 0.1633 and an RMI of 61.36%, better than VMI-FGSM + Our TSP Loss (0.1566 MSE, 54.74% RMI). These results indicate that simply appending the TSP loss to momentum variance sampling is insufficient to fully exploit spatial-temporal vulnerabilities, and the tailored search mechanism of STDSA yields substantially higher vulnerability exposure.

Under diverse input processing defenses (TV, Randomized Smoothing) and advanced defense training objectives (AT, KL, MSE Consistency, and the combined defense), STDSA shows greater persistence than VMI-FGSM + Our TSP Loss. Across almost all combinations of target models

Table A1: (Defended)-clean → undefended-adversarial → defended-adversarial MSE trajectories on ETTh1 under the STDSA attack. In the attack side, higher MSE (↑), higher RMI (↑), and lower AMR (↓) indicate better performance.

Defense	Attack Scheme	TimeBase	iTransformer	SparseTSF	FITS	LSINet	Model Macro Average
None	**STDSA**	MSE 0.3613→0.4184→**0.4184** RMI **15.80**% → AMR 0.00%	MSE 0.4080→0.4713→**0.4713** RMI **15.51**% → AMR 0.00%	MSE 0.3608→0.4221→**0.4221** RMI **16.99**% → AMR 0.00%	MSE 0.4384→0.5163→**0.5163** RMI **17.77**% → AMR 0.00%	MSE 0.3738→0.4464→**0.4464** RMI **19.42**% → AMR 0.00%	MSE 0.3885→0.4549→**0.4549** RMI **17.09**% → AMR 0.00%
	VMI-FGSM+Our TSP Loss	MSE 0.3613→0.4033→0.4033 RMI 11.62% → AMR 0.00%	MSE 0.4080→0.4587→0.4587 RMI 12.43% → AMR 0.00%	MSE 0.3608→0.4095→0.4095 RMI 13.50% → AMR 0.00%	MSE 0.4384→0.4967→0.4967 RMI 13.30% → AMR 0.00%	MSE 0.3738→0.4302→0.4302 RMI 15.09% → AMR 0.00%	MSE 0.3885→0.4397→0.4397 RMI 13.18% → AMR 0.00%
TV	**STDSA**	MSE 0.3607→0.4184→**0.4173** RMI **16.00**% → AMR 0.26%	MSE 0.4075→0.4713→**0.4702** RMI **15.66**% → AMR 0.23%	MSE 0.3609→0.4221→**0.4217** RMI **16.96**% → AMR 0.09%	MSE 0.4374→0.5163→**0.5148** RMI **18.04**% → AMR 0.29%	MSE 0.3738→0.4464→**0.4459** RMI **19.42**% → AMR 0.11%	MSE 0.3880→0.4549→**0.4540** RMI **17.24**% → AMR 0.20%
	VMI-FGSM+Our TSP Loss	MSE 0.3607→0.4033→0.4023 RMI 11.81% → AMR **0.25**%	MSE 0.4075→0.4587→0.4577 RMI 12.56% → AMR **0.22**%	MSE 0.3609→0.4095→0.4092 RMI 13.47% → AMR **0.07**%	MSE 0.4374→0.4967→0.4953 RMI 13.56% → AMR 0.28%	MSE 0.3738→0.4302→0.4298 RMI 15.09% → AMR **0.09**%	MSE 0.3880→0.4397→0.4388 RMI 13.32% → AMR **0.20**%
Randomized Smoothing	**STDSA**	MSE 0.3616→0.4184→**0.4141** RMI **15.71**% → AMR 1.03%	MSE 0.4052→0.4713→**0.4644** RMI **16.31**% → AMR 1.46%	MSE 0.3667→0.4221→**0.4240** RMI **15.11**% → AMR -0.45%	MSE 0.4257→0.5163→**0.4995** RMI **21.28**% → AMR 3.25%	MSE 0.3795→0.4464→**0.4475** RMI **17.63**% → AMR -0.25%	MSE 0.3877→0.4549→**0.4499** RMI **17.33**% → AMR 1.10%
	VMI-FGSM+Our TSP Loss	MSE 0.3616→0.4033→0.4002 RMI 11.53% → AMR **0.77**%	MSE 0.4052→0.4587→0.4526 RMI 13.20% → AMR **1.33**%	MSE 0.3667→0.4095→0.4121 RMI 11.67% → AMR **-0.63**%	MSE 0.4257→0.4967→0.4808 RMI 16.68% → AMR **3.20**%	MSE 0.3795→0.4302→0.4322 RMI 13.36% → AMR **-0.46**%	MSE 0.3877→0.4397→0.4356 RMI 13.41% → AMR **0.93**%
AT	**STDSA**	MSE 0.3558 → 0.4184 → **0.4117** RMI **17.59**% → AMR 1.60%	MSE 0.4283 → 0.4713 → **0.4849** RMI **10.04**% → AMR -2.89%	MSE 0.3486 → 0.4221 → **0.4047** RMI **21.08**% → AMR 4.12%	MSE 0.3539 → 0.5163 → **0.4115** RMI **45.89**% → AMR 20.30%	MSE 0.3709 → 0.4464 → **0.4409** RMI **20.36**% → AMR 1.23%	MSE 0.3715 → 0.4549 → **0.4307** RMI **22.45**% → AMR 5.32%
	VMI-FGSM+Our TSP Loss	MSE 0.3558 → 0.4033 → 0.3977 RMI 13.35% → AMR **1.39**%	MSE 0.4283 → 0.4587 → 0.4734 RMI 7.10% → AMR **-3.20**%	MSE 0.3486 → 0.4095 → 0.3933 RMI 17.47% → AMR **3.96**%	MSE 0.3539 → 0.4967 → 0.3993 RMI 40.35% → AMR **19.61**%	MSE 0.3709 → 0.4302 → 0.4256 RMI 15.99% → AMR **1.07**%	MSE 0.3715 → 0.4397 → 0.4179 RMI 18.36% → AMR **4.96**%
MSE Consistency	**STDSA**	MSE 0.3607 → 0.4184 → **0.4179** RMI **16.00**% → AMR 0.12%	MSE 0.4121 → 0.4713 → **0.4675** RMI **14.37**% → AMR 0.81%	MSE 0.3500 → 0.4221 → **0.4090** RMI **20.60**% → AMR 3.10%	MSE 0.3558 → 0.5163 → **0.4166** RMI **45.11**% → AMR 19.31%	MSE 0.3724 → 0.4464 → **0.4440** RMI **19.87**% → AMR 0.54%	MSE 0.3702 → 0.4549 → **0.4310** RMI **22.88**% → AMR 5.25%
	VMI-FGSM+Our TSP Loss	MSE 0.3607 → 0.4033 → 0.4045 RMI 11.81% → AMR **-0.30**%	MSE 0.4121 → 0.4587 → 0.4571 RMI 11.31% → AMR **0.35**%	MSE 0.3500 → 0.4095 → 0.3973 RMI 17.00% → AMR **2.98**%	MSE 0.3558 → 0.4967 → 0.4043 RMI 39.60% → AMR **18.60**%	MSE 0.3724 → 0.4302 → 0.4282 RMI 15.52% → AMR **0.46**%	MSE 0.3702 → 0.4397 → 0.4183 RMI 18.77% → AMR **4.87**%
KL Temporal Consistency	**STDSA**	MSE 0.3597 → 0.4184 → **0.4172** RMI **16.32**% → AMR 0.29%	MSE 0.4308 → 0.4713 → **0.4866** RMI **9.40**% → AMR -3.25%	MSE 0.3518 → 0.4221 → **0.4106** RMI **19.98**% → AMR 2.72%	MSE 0.3606 → 0.5163 → **0.4253** RMI **43.18**% → AMR 17.63%	MSE 0.3732 → 0.4464 → **0.4457** RMI **19.61**% → AMR 0.16%	MSE 0.3752 → 0.4549 → **0.4371** RMI **21.24**% → AMR 3.91%
	VMI-FGSM+Our TSP Loss	MSE 0.3597 → 0.4033 → 0.4057 RMI 12.12% → AMR **-0.60**%	MSE 0.4308 → 0.4587 → 0.4772 RMI 6.48% → AMR **-4.03**%	MSE 0.3518 → 0.4095 → 0.3995 RMI 16.40% → AMR **2.44**%	MSE 0.3606 → 0.4967 → 0.4116 RMI 37.74% → AMR **17.13**%	MSE 0.3732 → 0.4302 → 0.4295 RMI 15.27% → AMR 0.16%	MSE 0.3752 → 0.4397 → 0.4247 RMI 17.19% → AMR **3.41**%
AT + MSE Consistency + TV	**STDSA**	MSE 0.3568 → 0.4184 → **0.4118** RMI **17.26**% → AMR 1.58%	MSE 0.3992 → 0.4713 → **0.4526** RMI **18.06**% → AMR 3.97%	MSE 0.3492 → 0.4221 → **0.4049** RMI **20.88**% → AMR 4.07%	MSE 0.3535 → 0.5163 → **0.4105** RMI **46.05**% → AMR 20.49%	MSE 0.3708 → 0.4464 → **0.4401** RMI **20.39**% → AMR 1.41%	MSE 0.3659 → 0.4549 → **0.4240** RMI **24.32**% → AMR 6.79%
	VMI-FGSM+Our TSP Loss	MSE 0.3568 → 0.4033 → 0.4017 RMI 13.03% → AMR **0.40**%	MSE 0.3992 → 0.4587 → 0.4417 RMI 14.90% → AMR **3.71**%	MSE 0.3492 → 0.4095 → 0.3934 RMI 17.27% → AMR **3.93**%	MSE 0.3535 → 0.4967 → 0.3984 RMI 40.51% → AMR **19.79**%	MSE 0.3708 → 0.4302 → 0.4249 RMI 16.02% → AMR **1.23**%	MSE 0.3659 → 0.4397 → 0.4120 RMI 20.17% → AMR **6.30**%

Table A2: (Defended)-clean → undefended-adversarial → defended-adversarial MSE trajectories on Exchange under the STDSA attack. In the attack side, higher MSE (↑), higher RMI (↑), and lower AMR (↓) indicate better performance.

Defense	Attack Scheme	TimeBase	iTransformer	SparseTSF	FITS	LSINet	Model Macro Average
None	**STDSA**	MSE 0.1174 → 0.1857 → **0.1857** RMI **58.18**% → AMR 0.00%	MSE 0.1115 → 0.1560 → **0.1560** RMI **39.91**% → AMR 0.00%	MSE 0.1016 → 0.1739 → **0.1739** RMI **71.16**% → AMR 0.00%	MSE 0.0873 → 0.1538 → 0.1538 RMI 76.17% → AMR 0.00%	MSE 0.0882 → 0.1470 → **0.1470** RMI **66.67**% → AMR 0.00%	MSE 0.1012 → 0.1633 → **0.1633** RMI **61.36**% → AMR 0.00%
	VMI-FGSM+Our TSP Loss	MSE 0.1174 → 0.1779 → 0.1779 RMI 51.53% → AMR 0.00%	MSE 0.1115 → 0.1435 → 0.1435 RMI 28.70% → AMR 0.00%	MSE 0.1016 → 0.1657 → 0.1657 RMI 63.09% → AMR 0.00%	MSE 0.0873 → 0.1573 → **0.1573** RMI **80.18**% → AMR 0.00%	MSE 0.0882 → 0.1387 → 0.1387 RMI 57.26% → AMR 0.00%	MSE 0.1012 → 0.1566 → 0.1566 RMI 54.74% → AMR 0.00%
TV	**STDSA**	MSE 0.1167 → 0.1857 → **0.1848** RMI **59.13**% → AMR **0.48**%	MSE 0.1115 → 0.1560 → **0.1555** RMI **39.91**% → AMR 0.32%	MSE 0.1011 → 0.1739 → **0.1731** RMI **72.01**% → AMR **0.46**%	MSE 0.0872 → 0.1538 → 0.1538 RMI 76.38% → AMR **0.00**%	MSE 0.0882 → 0.1470 → **0.1474** RMI **66.67**% → AMR **-0.27**%	MSE 0.1010 → 0.1633 → **0.1629** RMI **61.68**% → AMR 0.24%
	VMI-FGSM+Our TSP Loss	MSE 0.1167 → 0.1779 → 0.1770 RMI 52.44% → AMR 0.51%	MSE 0.1115 → 0.1435 → 0.1436 RMI 28.70% → AMR **-0.07**%	MSE 0.1011 → 0.1657 → 0.1648 RMI 63.90% → AMR 0.54%	MSE 0.0872 → 0.1573 → **0.1571** RMI **80.39**% → AMR 0.13%	MSE 0.0882 → 0.1387 → 0.1388 RMI 57.26% → AMR -0.07%	MSE 0.1010 → 0.1566 → 0.1563 RMI 55.05% → AMR **0.19**%
Randomized Smoothing	**STDSA**	MSE 0.1155 → 0.1857 → **0.1832** RMI **60.78**% → AMR **1.35**%	MSE 0.1114 → 0.1560 → **0.1541** RMI **40.04**% → AMR 1.22%	MSE 0.1001 → 0.1739 → **0.1715** RMI **73.73**% → AMR **1.38**%	MSE 0.0868 → 0.1538 → 0.1530 RMI 77.19% → AMR **0.52**%	MSE 0.0883 → 0.1470 → **0.1466** RMI **66.48**% → AMR 0.27%	MSE 0.1004 → 0.1633 → **0.1617** RMI **62.65**% → AMR 0.98%
	VMI-FGSM+Our TSP Loss	MSE 0.1155 → 0.1779 → 0.1754 RMI 54.03% → AMR 1.41%	MSE 0.1114 → 0.1435 → 0.1438 RMI 28.82% → AMR **-0.21**%	MSE 0.1001 → 0.1657 → 0.1634 RMI 65.53% → AMR 1.39%	MSE 0.0868 → 0.1573 → **0.1562** RMI **81.22**% → AMR 0.70%	MSE 0.0883 → 0.1387 → 0.1390 RMI 57.08% → AMR **-0.22**%	MSE 0.1004 → 0.1566 → 0.1555 RMI 55.98% → AMR **0.70**%
AT	**STDSA**	MSE 0.1059 → 0.1857 → **0.1551** RMI **75.35**% → AMR **16.48**%	MSE 0.1139 → 0.1560 → **0.1437** RMI **36.96**% → AMR 7.88%	MSE 0.0999 → 0.1739 → **0.1483** RMI **74.07**% → AMR **14.72**%	MSE 0.0852 → 0.1538 → **0.1291** RMI 80.52% → AMR **16.06**%	MSE 0.0954 → 0.1470 → **0.1341** RMI **54.09**% → AMR 8.78%	MSE 0.1001 → 0.1633 → **0.1420** RMI **63.14**% → AMR **13.04**%
	VMI-FGSM+Our TSP Loss	MSE 0.1059 → 0.1779 → 0.1436 RMI 67.99% → AMR 19.28%	MSE 0.1139 → 0.1435 → 0.1399 RMI 25.99% → AMR **2.51**%	MSE 0.0999 → 0.1657 → 0.1363 RMI 65.87% → AMR 17.74%	MSE 0.0852 → 0.1573 → 0.1167 RMI **84.62**% → AMR 25.81%	MSE 0.0954 → 0.1387 → 0.1292 RMI 45.39% → AMR **6.85**%	MSE 0.1001 → 0.1566 → 0.1332 RMI 56.44% → AMR 14.94%
MSE Consistency	**STDSA**	MSE 0.1048 → 0.1857 → **0.1539** RMI **77.19**% → AMR **17.12**%	MSE 0.1003 → 0.1560 → **0.1341** RMI **55.53**% → AMR 14.04%	MSE 0.0999 → 0.1739 → **0.1499** RMI **74.07**% → AMR **13.80**%	MSE 0.0840 → 0.1538 → **0.1288** RMI 83.10% → AMR **16.25**%	MSE 0.0918 → 0.1470 → **0.1351** RMI **60.13**% → AMR 8.10%	MSE 0.0962 → 0.1633 → **0.1403** RMI **69.75**% → AMR **14.08**%
	VMI-FGSM+Our TSP Loss	MSE 0.1048 → 0.1779 → 0.1425 RMI 69.75% → AMR 19.90%	MSE 0.1003 → 0.1435 → 0.1261 RMI 43.07% → AMR **12.13**%	MSE 0.0999 → 0.1657 → 0.1389 RMI 65.87% → AMR 16.17%	MSE 0.0840 → 0.1573 → 0.1172 RMI **87.26**% → AMR 25.49%	MSE 0.0918 → 0.1387 → 0.1297 RMI 51.09% → AMR **6.49**%	MSE 0.0962 → 0.1566 → 0.1309 RMI 62.79% → AMR 16.41%
KL Temporal Consistency	**STDSA**	MSE 0.1069 → 0.1857 → **0.1560** RMI **73.71**% → AMR **15.99**%	MSE 0.1154 → 0.1560 → **0.1539** RMI **35.18**% → AMR 1.35%	MSE 0.0998 → 0.1739 → **0.1535** RMI **74.25**% → AMR **11.73**%	MSE 0.0844 → 0.1538 → 0.1429 RMI 82.23% → AMR **7.09**%	MSE 0.0904 → 0.1470 → 0.1499 RMI **62.61**% → AMR -1.97%	MSE 0.0994 → 0.1633 → **0.1512** RMI **64.29**% → AMR 7.41%
	VMI-FGSM+Our TSP Loss	MSE 0.1069 → 0.1779 → 0.1442 RMI 66.42% → AMR 18.94%	MSE 0.1154 → 0.1435 → 0.1435 RMI 24.35% → AMR **0.00**%	MSE 0.0998 → 0.1657 → 0.1428 RMI 66.03% → AMR 13.82%	MSE 0.0844 → 0.1573 → **0.1444** RMI **86.37**% → AMR 8.20%	MSE 0.0904 → 0.1387 → **0.1521** RMI 53.43% → AMR **-9.66**%	MSE 0.0994 → 0.1566 → 0.1454 RMI 57.55% → AMR **7.15**%
AT + MSE Consistency + TV	**STDSA**	MSE 0.1045 → 0.1857 → **0.1534** RMI **77.70**% → AMR **17.39**%	MSE 0.1334 → 0.1560 → **0.1641** RMI **16.94**% → AMR -5.19%	MSE 0.0993 → 0.1739 → **0.1473** RMI **75.13**% → AMR **15.30**%	MSE 0.0850 → 0.1538 → **0.1289** RMI 80.94% → AMR **16.19**%	MSE 0.0959 → 0.1470 → **0.1333** RMI **53.28**% → AMR 9.32%	MSE 0.1036 → 0.1633 → **0.1454** RMI **57.63**% → AMR **10.96**%
	VMI-FGSM+Our TSP Loss	MSE 0.1045 → 0.1779 → 0.1420 RMI 70.24% → AMR 20.18%	MSE 0.1334 → 0.1435 → 0.1554 RMI 7.57% → AMR **-8.29**%	MSE 0.0993 → 0.1657 → 0.1353 RMI 66.87% → AMR 18.35%	MSE 0.0850 → 0.1573 → 0.1166 RMI **85.06**% → AMR 25.87%	MSE 0.0959 → 0.1387 → 0.1273 RMI 44.63% → AMR **8.22**%	MSE 0.1036 → 0.1566 → 0.1353 RMI 51.16% → AMR 13.60%

and datasets, STDSA maintains higher post-defense MSE trajectory endpoints (defended-clean → undefended-adversarial → defended-adversarial). For example, under the combined defense (AT + MSE Consistency + TV) on Weather, STDSA preserves a macro average defended MSE of 0.2045 versus 0.1950 for VMI-FGSM + Our TSP Loss. On backbones ranging from iTransformer and frequency-domain models (FITS) to lightweight linear networks (SparseTSF and LSINet), perturbations generated by STDSA appear more persistent against the defenses. This comparison verifies that the gain of STDSA stems not only from the TSP loss formulation, but also from its intrinsic ability to craft structural perturbations that survive temporal smoothing and adversarial alignment defenses.

A.6.3 Comparison with Random Perturbations

To verify that the disruption caused by STDSA does not only result from injecting random noise, we compare STDSA with two non-optimized Gaussian perturbation baselines on ETTh1. Both baselines are generated by PCG64 without surrogate models, target-model gradients, prediction queries, or iterative optimization. Gaussian-ℓ_∞ is normalized per sample to match the perturbation budget of STDSA ($\epsilon = 0.1$), while Gaussian-SNR is globally scaled to match the SNR of STDSA (21.52dB). The same Gaussian random direction is used for both baselines. All methods are evaluated on the same test samples.

Table A3: Relative MSE increase (%) caused by STDSA and non-optimized Gaussian perturbations on ETTh1 (upper), Weather (middle), and Exchange (bottom). Gaussian-ℓ_∞ matches the ℓ_∞ budget of STDSA, while Gaussian-SNR matches its global SNR. A higher value indicates a stronger disruption.

Target model	STDSA	Gaussian-ℓ_∞	Gaussian-SNR
ETTh1			
TimeBase	**15.80**	0.04	0.48
iTransformer	**15.51**	0.03	0.44
SparseTSF	**16.99**	0.03	0.28
FITS	**17.77**	0.04	0.71
LSINet	**19.42**	0.03	0.35
Average	**17.09**	0.03	0.45
Weather			
TimeBase	**24.41**	0.33	5.46
iTransformer	**27.27**	-1.05	-1.97
SparseTSF	**31.84**	0.02	0.19
FITS	**32.76**	0.02	0.25
LSINet	**33.93**	0.08	1.11
Average	**30.08**	-0.12	1.01
Exchange			
TimeBase	**58.18**	0.66	9.95
iTransformer	**39.91**	0.02	0.88
SparseTSF	**71.16**	0.61	9.20
FITS	**76.17**	0.51	7.50
LSINet	**66.67**	0.25	3.69
Average	**61.36**	0.41	6.24

Table A3 shows that STDSA causes substantially greater forecasting disruption than random Gaussian perturbations. This significant performance gap indicates that random Gaussian perturbations cannot make the most of the attack budget as STDSA does. On ETTh1, Gaussian-ℓ_∞ increases the MSE by only 0.03% on average, while STDSA causes an average RMI of 17.09%, under the same perturbation budget. On Weather, Gaussian-ℓ_∞ even **reduces** the MSE by 0.12%, while STDSA increases the MSE of every target model by between 24.41% and 33.93%, achieving a five-target average increase of 30.08%. On iTransformer, both Gaussian controls produce negative MSE changes, which means that the corresponding random perturbations slightly reduce the forecasting error. This can occur because a random perturbation may occasionally move an input toward a locally lower-error region, which indicates that the attack is ineffective. On Exchange, Gaussian-ℓ_∞ causes an average RMI of only 0.41%, while STDSA causes a high average RMI of 61.36%. Besides, STDSA

causes significantly higher disruption on Exchange compared to Weather and ETTh1, which indicates that financial or exchange rate time series are particularly sensitive to targeted perturbations.

When matched to the global SNR of STDSA, Gaussian-SNR causes average RMIs of 0.45%, 1.01%, and 6.24% on ETTh1, Weather, and Exchange, respectively. It is worth noting that the resulting Gaussian perturbation has an actual ℓ_∞ value of 0.544, which is substantially larger than the ℓ_∞ value of STDSA (0.100), but the former remains much less effective. Even on its highest-impact target (TimeBase on Exchange at 9.95%), Gaussian-SNR fails to reach even a 10% RMI despite matching the total signal power of STDSA. Comparing Gaussian-ℓ_∞ to Gaussian-SNR shows that simply increasing total noise power yields modest gains (e.g., from 0.41% to 6.24% on Exchange). These results suggest that the broad effectiveness of STDSA arises from its optimized spectral-temporal perturbation direction rather than perturbation magnitude alone.

A.7 Ablation Study

We conduct an ablation study for STDSA on ETTh1 with the same five target models. All ablated variants use the same test set, attack budget ($\epsilon = 0.1$), step size ($\eta = 0.01$), 200 attack iterations, and evaluation protocol. As shown in Table A4, Full STDSA achieves a mean RMI of 17.09% across the five targets. `Periodic Pathway Only` reduces the mean RMI to 15.75%, which indicates that the direct pathway provides complementary gradient information. Although `Direct Pathway Only` achieves a comparable mean RMI of 17.07%, its cross-target standard deviation (STD) reaches 4.70, compared with 1.42 for Full STDSA. Besides, it performs well on FITS and LSINet, but performs substantially worse on the other three targets. Hence the two structural pathways of STDSA improves cross-model coverage and consistency. Removing LPF results in a mean RMI of 16.99%, which indicates that spectral processing still provides a modest benefit. To summarize, the full STDSA is the best choice considering both performance and stability.

Table A4: Relative MSE increase (%) for different ablated variants of STDSA on ETTh1. A higher value indicates a stronger disruption.

Variant	TimeBase	iTransformer	SparseTSF	FITS	LSINet	Mean	STD
Full STDSA	15.80	15.51	16.99	17.77	19.42	**17.09**	1.42
`Without LPF`	15.17	16.07	17.40	17.01	19.33	16.99	1.40
`Periodic Pathway Only`	15.75	14.62	16.69	15.08	16.60	15.75	0.82
`Direct Pathway Only`	12.22	14.00	13.64	21.88	23.60	17.07	4.70

A.8 Hyperparameter Sensitivity Analysis

To evaluate the robustness and stability of STDSA, we conduct a sensitivity analysis for four key hyperparameters (see Figure A1): the mixing parameter w_1 (default 0.9), the frequency cutoff budget K_{keep} (default 12), the spectral blending parameter α (default 0.75), and the perturbation budget ϵ (default 0.1). All evaluations measure the RMI (%) alongside STD across different target models.

The mixing parameter w_1 determines the structural component weighting. The RMI reaches its peak at mid-range values ($w_1 = 0.60$, reaching an RMI of 18.4%) but shows higher variance across different target models. Pushing w_1 beyond 0.60 suppresses RMI down to a minimum of 15.8% at $w_1 = 1.00$. The default configuration ($w_1 = 0.90$) achieves a favorable trade-off that maintains low STD with a 17.09% RMI.

The frequency cutoff budget K_{keep} shows the effect of spectral preservation on model outputs. For small $K_{keep} \in [1, 4]$, RMI remains relatively low ($\approx 15.5\% \sim 15.7\%$). As K_{keep} increases from 4 to 12, the RMI also increases, eventually plateauing at $\approx 17.00\% \sim 17.09\%$ for $K_{keep} \geqslant 12$. Hence the default choice of $K_{keep} = 12$ captures necessary spectral context while operating in the stable plateau regime.

The spectral blending parameter α shows insensitivity across its entire range $[0.00, 1.00]$. The RMI remains tightly constrained within a narrow window between 16.8% and 17.2%. Our main experimental setting ($\alpha = 0.75$) yields a 17.09% increase, which confirms that STDSA remains robust to adjustments in α.

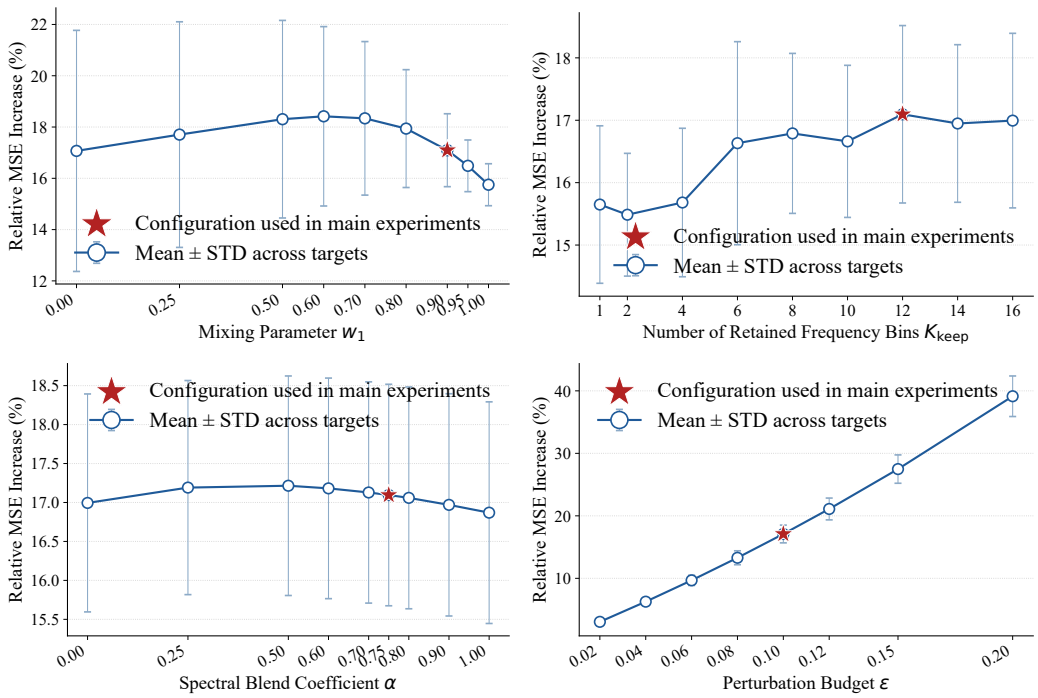

Figure A1: Sensitivity analysis for four key hyperparameters of STDSA.

In contrast to the algorithmic hyperparameter dynamics, scaling the perturbation budget ϵ shows a strict monotonic and near-linear increase in RMI. As ϵ increases from 0.02 to 0.20, the RMI rises from 3.0% up to 39.2%, which indicates that the disruption of STDSA scales well along with the perturbation budget. To ensure stealthiness of STDSA, $\epsilon = 0.1$ is an appropriate choice with an RMI of 17.09% and an SNR of 20.27dB.

AI Use Statement

In this work, we used generative AI tools only for grammar correction and minor language edits to improve readability. We have not used generative AI tools for other tasks. We have reviewed all AI-assisted work, and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.