Rubato: Signature Attention for Irregular Multivariate Time Series Forecasting

Jintao Yang
East China Normal University
jtyang@stu.ecnu.edu.cn

Meng Wang
Xi'an Polytechnic University
wangmeng@xpu.edu.cn

Chenjuan Guo*
East China Normal University
cjguo@dase.ecnu.edu.cn

Bin Yang
East China Normal University
byang@dase.ecnu.edu.cn

Abstract

Irregular multivariate time series (IMTS) are sampled at non-uniform timestamps and asynchronous across channels, their dependencies are shaped by time gaps, local ordering, and channel-specific sampling patterns. Building a faithful attention mechanism for IMTS exposes two limitations of existing work. First, current irregular-time attention methods typically regularize raw inputs into interpolated, aligned, or continuous-time-embedded forms before applying attention, which blurs exactly these irregular cues that determine how two observations relate. Second, beyond representation, standard dot-product attention scores pairs through learned token projections with no intrinsic tie to the shape of the underlying series, and an attention that does reflect this shape becomes costly to compute in high-channel IMTS. We propose RUBATO, which reads two views directly from the raw irregular input: per-channel local dynamics, and a path signature that summarizes the joint multi-channel series. Attention in RUBATO then scores pairs by a signature-kernel measure of series similarity rather than by a generic dot product, and a structured random projection (SORF) keeps this comparison affordable when the channel count is large. We prove that the structured projection introduces only a controlled bias that vanishes as the number of channels grows, and experiments on multiple IMTS benchmarks show that RUBATO consistently outperforms strong baselines. Code is available at https://github.com/decisionintelligence/Rubato.

1 Introduction

Multivariate time series forecasting [25, 27, 26, 29, 40, 17, 4, 5] is central to domains such as finance [34], traffic [43], and electricity load [24]. Unlike regular time series, *irregular multivariate time series* (IMTS) are observed at non-uniform timestamps and sampled asynchronously across channels, so they no longer satisfy the uniform-sampling assumption that most forecasting models rely on. A model for IMTS must therefore handle uneven observation intervals within each channel and, at the same time, capture dependencies between asynchronously observed channels.

Existing IMTS methods [28, 18] span several paradigms (Figure 1). ODE-based methods [31, 9, 13, 33, 2] treat an irregular sequence as a dynamical system evolving along the time axis. Irregular timestamps are absorbed naturally, but information propagates step by step in temporal order, so two observations that are far apart in time can only influence each other indirectly through the chain of

*Corresponding author.

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

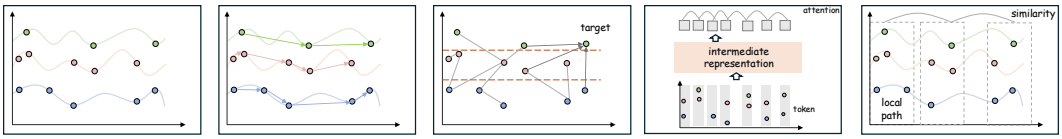


Figure 1: IMTS modeling paradigms. **(a) Raw IMTS**: Non-uniform timestamps, asynchronous across channels. **(b) ODE-based**: A latent state evolves along the time axis, so distant observations can only influence each other through the chain of intermediate states. **(c) Graph-based**: Pre-defined edges and message passing decide interactions independently of observation content. **(d) Attention-based**: Raw inputs are first regularized (interpolation/alignment/embeddings), blurring time gaps and asynchronous sampling. **(e) RUBATO (ours)**: A path signature, computed directly on the raw irregular input, drives attention through a signature kernel.

intermediate states. Graph-based methods [42, 45] connect observations through pre-defined edges and propagate information by message passing; this exploits useful structural priors, but which observations interact, and how strongly, is fixed by the graph construction rather than determined by the content of the observations themselves. Attention is, in principle, better suited to IMTS: dependencies are cross-time, cross-variable, and strongly context-dependent, and attention can establish a direct interaction between any pair of observations. The problem with current irregular-time attention [35, 11] lies one step earlier: raw observations are first regularized into interpolated, aligned, or continuous-time-embedded forms before attention is applied. As a result, the time gaps, asynchronous sampling, and local evolution structure of the original sequence are blurred, and the similarity computed by attention no longer reflects how the original observations are arranged in time.

Turning attention into a faithful mechanism for IMTS requires solving two progressive challenges.

Challenge 1: Representing irregular observations without distorting their temporal structure. The problem is that current irregular-time attention sees the input only after a regularization step (interpolation, alignment, or continuous-time embedding), so time gaps, local ordering, and asynchronous cross-channel variation are smoothed out before they ever reach the attention mechanism, even though they are exactly the structure that should determine how observations relate.

To avoid this loss, we build representations directly from the raw irregular observations, without first projecting them onto a uniform time grid. Each input is encoded from two complementary views: per-channel local dynamics, and a *path signature* [21, 7] of the joint observation path. The path signature is a mathematical object from rough path theory that summarizes the shape and ordering of the joint multi-channel series, viewed as a single path. Because both views are computed on the raw input, the time gaps, ordering, and cross-channel asynchrony of the original sequence are preserved for downstream dependency modeling.

Challenge 2: Comparing series similarity affordably for dependency modeling. Even with such representations in hand, standard QK attention still decides interactions from dot products of learned token projections, with no intrinsic tie to the series shape and ordering preserved in the signature. A similarity that does reflect this structure, however, is expensive to compute when the channel count is large: explicit comparisons between series typically require pairwise evaluations whose cost grows linearly with the channel count under dense projections.

Building on this, we construct an attention mechanism in which interaction strengths are decided by how similar two series are, rather than by conventional dot products. We make this comparison computationally tractable in two steps: (i) we summarize each series through its path signature, the same compact summary used in our representation step, so that two series can be compared through an inner product of their summaries; and (ii) we approximate this inner product with Structured Orthogonal Random Features (SORF) [44], a structured random projection whose cost grows more favorably with the number of channels than dense alternatives. Together, these give RUBATO: it extracts per-channel local dynamics and a joint-path signature summary from the raw irregular input (Challenge 1), models cross-time and cross-variable dependencies through signature-kernel similarity (Challenge 2), and produces forecasts at arbitrary future times via a time-conditioned decoder.

Contributions.

- We propose RUBATO, an attention-based architecture for IMTS forecasting that unifies signature-based input representation, signature-kernel attention, and time-conditioned decoding in a single framework, enabling more natural modeling of cross-time and cross-variable dependencies under irregular sampling.
- We prove that, inside the random-feature approximation of the signature kernel adopted in our architecture (see Section 4), replacing dense random Fourier projections with Structured Orthogonal Random Features (SORF) introduces only an $O(\log d_H / d_H)$ bias, where d_H is the Hadamard-padded block dimension and grows with the channel count C; this bias therefore asymptotically vanishes as the number of channels grows, providing a theoretical basis for this efficient approximation.
- We validate the effectiveness of our approach on multiple IMTS benchmarks. RUBATO consistently outperforms strong existing baselines in forecasting performance, and the SORF-based feature map matches or improves over the standard dense baseline in accuracy across all tested dimensions while offering asymptotically cheaper projection cost.

2 Related Work

Irregular time series. GRU-D [3] augments gated recurrent units with decay to handle variable inter-observation gaps. ODE-based approaches treat the irregular sequence as a dynamical system evolving along the time axis: Latent ODE [31], GRU-ODE-Bayes [9], neural controlled differential equations [13], CRU [33], and neural flows [2] model dynamics via neural ODEs or their variants. While irregular timestamps are handled naturally, information still propagates step by step along the time axis, making long-range cross-observation interactions hard to express. Graph-based methods connect observations through predefined rules and propagate information via message passing, exploiting structural priors: GraFITi [42] casts forecasting as edge-weight prediction on a sparse bipartite graph, t-PatchGNN [45] introduces transformable patching for irregular sequences, HiPatch [20] stacks hierarchical patch GNN layers across temporal scales, and HyperIMTS [16] models inter-observation relations on a hypergraph; nevertheless, which observations interact is largely decided by graph construction rather than by the content of the observations themselves. More recent approaches further explore alternative inductive biases: APN [19] returns to a simple time-aware patching baseline that aggregates observations within adaptive temporal windows, and KAFNet [46] revitalizes canonical pre-alignment by mapping irregular observations onto a shared regular grid before applying frequency-domain mixing.

Signature methods. Path signatures from rough path theory [21] provide canonical sequential features. Király and Oberhauser [14] formalized the signature kernel; Salvi et al. [32] showed it satisfies a Goursat PDE. Random Fourier Signature Features (RFSF) [37] scale signature kernels via random projections, but inherit the $O(d\tilde{d})$ cost of dense RFF. Rough Transformers [23] concatenate local and global signatures (or log-signatures) into multi-view tokens and apply dot-product attention with independently learned query/key maps. Both methods exploit continuous-path structure under irregular sampling; RUBATO instead constructs symmetric, cosine-normalized pre-softmax scores using a shared per-head projection of randomized joint value-mask-time signature features. For scalability, RUBATO combines SORF with TRP to obtain width $D_{\mathrm{sig}} = 1 + Mr$ at fixed order M and compression width r, avoiding explicit joint-signature coordinates. Rough Transformers also discuss log-signatures, per-channel signatures, and randomized signatures as remedies for dimensionality; our distinction lies in kernel-induced attention and its SORF approximation.

Random feature approximations. RFF [30] approximate shift-invariant kernels via random projections. Yu et al. [44] proposed ORF/SORF with an empirical variance reduction for RBF kernels and $O(\tilde{d} \log d)$ cost via the Walsh–Hadamard transform, building on Ailon and Chazelle [1]. Fastfood [15] proposed a related Hadamard-based construction. SORF has been studied for RBF kernels but *not* for signature kernels; bridging this gap is our central theoretical contribution. Our goal is not merely to accelerate kernel approximation in isolation, but to show that SORF can be used inside signature-kernel feature recursion without losing theoretical control: Theorem 2 provides the first bias bound for SORF signature kernel approximation, and the resulting feature map preserves the accuracy of the downstream forecasting architecture (Section 5.2).

Irregular attention. mTAN [35] attends over learned continuous-time embeddings, Set Functions for Time Series [11] aggregate (time, channel, value) tuples with permutation-invariant attention,

and PrimeNet [8] pretrains an attention encoder with time-aware masking. In all of these, raw observations are first converted into embedded or set forms, so time gaps, asynchronous sampling, and local evolution are not directly used to decide inter-observation similarity. RUBATO instead derives attention scores from a random-feature approximation of the *signature kernel*, measuring path-level similarity directly on the raw irregular input.

3 Preliminaries

3.1 Problem Setup

We consider a multivariate time series with C channels observed at potentially different, irregular time points. An observation mask $\boldsymbol{M} \in \{0,1\}^{L_{\text{hist}} \times C}$ indicates which channels are observed at each step. Given historical irregular multivariate time series $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{L_{\text{hist}}})$ with timestamps $(t_1, \ldots, t_{L_{\text{hist}}})$ and mask $\boldsymbol{M}$, the goal is to predict $\hat{\boldsymbol{x}} \in \mathbb{R}^{L_{\text{tar}} \times C}$ at target times $(t_{L_{\text{hist}}+1}, \ldots, t_{L_{\text{hist}}+L_{\text{tar}}})$. The signature constructions below operate on a *path*, i.e., a discrete-time sequence of vectors in $\mathbb{R}^d$. An IMTS is not directly such a sequence, since not every step carries a complete $\mathbb{R}^C$ observation. We therefore form a *joint path* of dimension $p{=}2C{+}1$ that stacks values, mask, and time coordinate at each step, encoding irregularity as part of the path geometry rather than as missing data.

Notation. $d_H = 2^{\lceil \log_2 p \rceil}$ is the Hadamard-padded dimension used by SORF; D_{rf} is the number of random Fourier frequencies; $r = d_{\text{feat}}$ is the TRP-compressed per-level feature dimension; $D_{\text{sig}} = 1 + Mr$ is the final signature feature dimension.

3.2 Signature kernel: a feature-map view of path similarity

RUBATO scores attention pairs by signature-kernel similarity rather than by token dot product, so we briefly recall the construction we will use. For discrete paths $\boldsymbol{x}, \boldsymbol{y}$ in $\mathbb{R}^d$ with a base kernel k, the m-th order signature kernel is defined recursively [14]:

$$K^{(0)}_{s,t} \equiv 1, \qquad K^{(m)}_{s,t} = \sum_{i \le s,\, j \le t} K^{(m-1)}_{i-1,j-1}\, \delta^2_{i,j} k, \tag{1}$$

with $\delta^2_{i,j} k = k(\boldsymbol{x}_i, \boldsymbol{y}_j) - k(\boldsymbol{x}_i, \boldsymbol{y}_{j-1}) - k(\boldsymbol{x}_{i-1}, \boldsymbol{y}_j) + k(\boldsymbol{x}_{i-1}, \boldsymbol{y}_{j-1})$ and $|K^{(m)}| \le L_k^m \omega^m / (m!)^2$, where $L_k = \sup \|\nabla^2 k\|_2$ and $\omega = V_{\boldsymbol{x}} V_{\boldsymbol{y}}$ is the 1-variation product.

Direct evaluation of Equation 1 is $O(MS^2T^2)$ per pair, prohibitive when used inside attention. Tóth et al. [37] show that approximating k by a random feature map $\tilde{k}(\boldsymbol{u}, \boldsymbol{v}) = \langle \varphi(\boldsymbol{u}), \varphi(\boldsymbol{v}) \rangle$ turns the recursion into a single inner product $\tilde{K}_{\le M} = \langle \boldsymbol{\psi}_{\boldsymbol{x}}, \boldsymbol{\psi}_{\boldsymbol{y}} \rangle$ of per-path signature features—this is the RFSF construction, and it reduces the per-path cost to $O((S+T)Mr)$ with the TRP compression of Section 4.4. RUBATO adopts RFSF but substitutes SORF for the dense RFF projection, with independent SORF matrices $\boldsymbol{W}^{(m)}$ across levels (an inter-level independence the bias analysis in Section 4.4 requires).

3.3 Random and structured Fourier features

The base random feature map of Section 3 above is what RFSF expects; RUBATO replaces its dense version by a structured one to keep the joint-path feature stage affordable as C grows. For the Gaussian kernel, Bochner's theorem gives $k(\boldsymbol{u}, \boldsymbol{v}) = \mathbb{E}_{\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \sigma^{-2}\boldsymbol{I})}[\cos(\boldsymbol{w}^\top(\boldsymbol{u} - \boldsymbol{v}))]$, realized by random Fourier features (RFF) [30] with D_{rf} Gaussian rows. RFF is unbiased but costs $O(d_H D_{\text{rf}})$ per point, dominated by a dense matrix–vector product.

SORF [44] replaces the dense matrix with

$$\boldsymbol{W}_{\text{SORF}} = \tfrac{\sqrt{d_H}}{\sigma}\, \boldsymbol{H}\boldsymbol{D}_1 \boldsymbol{H}\boldsymbol{D}_2 \boldsymbol{H}\boldsymbol{D}_3 \in \mathbb{R}^{d_H \times d_H}, \tag{2}$$

where $\boldsymbol{H}$ is the normalized Walsh–Hadamard matrix and $\boldsymbol{D}_{1,2,3}$ are diagonal Rademacher matrices. Three fast Walsh–Hadamard transforms reduce the per-block cost from $O(d_H^2)$ to $O(d_H \log d_H)$, with total cost $O(D_{\text{rf}} \log d_H)$ when stacking $\lceil D_{\text{rf}}/d_H \rceil$ blocks. Unlike RFF, SORF introduces a structured bias independent of D_{rf}, which we bound in Section 4.4; increasing D_{rf} reduces variance but not this floor.

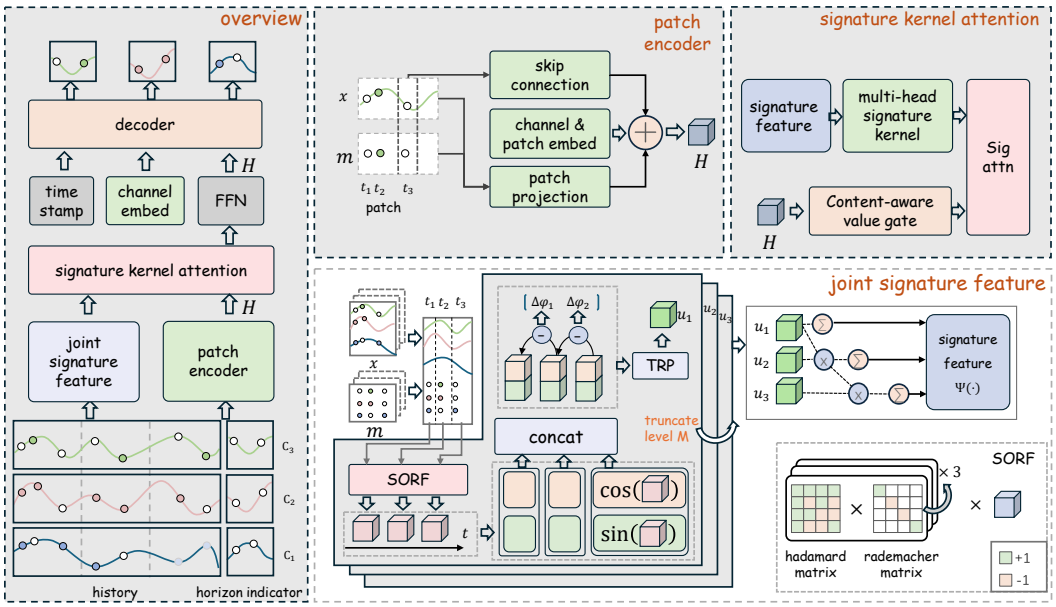


Figure 2: Architecture of RUBATO. The joint signature feature module produces SORF-based path-signature features $\boldsymbol{\Psi}$ from the multivariate joint path, while the patch encoder produces per-channel representations $\boldsymbol{H}$. Signature-kernel attention refines $\boldsymbol{H}$ using $\boldsymbol{\Psi}$, and the decoder cross-attends from time-conditioned queries to $\boldsymbol{H}$ to predict at arbitrary future timestamps.

4 The RUBATO Architecture

RUBATO processes an irregular multivariate time series through three stages (Figure 2): (1) dual-path tokenization producing per-channel patch tokens and a joint-path signature feature, (2) signature-kernel attention with per-head learned projections, cosine normalization, and content-dependent value gating, and (3) a time-conditioned decoder that generates predictions at arbitrary future time points. Section 4.4 states the bias bound for the SORF signature kernel approximation used in (1).

4.1 Dual-Path Tokenization

Given an input batch of irregular multivariate time series with B samples, C channels, and $L = L_{\text{hist}} + L_{\text{tar}}$ stacked time steps (history concatenated with forecast horizon), two parallel paths extract complementary representations.

4.1.1 Joint Signature Feature

We partition the joint history-and-horizon sequence into N non-overlapping patches of length $P = \lceil L/N \rceil$ along time. To expose asynchronous cross-channel temporal structure to the signature feature map, we form for each patch n a p-dimensional joint path ($p = 2C+1$) across the P steps $s = 1, \ldots, P$ within the patch, by concatenating observation values, masks, and a shared time coordinate:

$$\boldsymbol{p}_n^{(s)} = [\underbrace{\boldsymbol{x}_{1,n}^{(s)}, \ldots, \boldsymbol{x}_{C,n}^{(s)}}_{\text{values}}, \underbrace{m_{1,n}^{(s)}, \ldots, m_{C,n}^{(s)}}_{\text{masks}}, \underbrace{t_n^{(s)}}_{\text{time}}] \in \mathbb{R}^p. \tag{3}$$

Because the mask channels in Equation 3 already mark which steps carry observations, the signature computation needs no auxiliary step mask or forward-fill preprocessing. For each signature level $m \in \{1, \ldots, M\}$ and step s, we apply the SORF random-feature map of Section 3 to the path increment $\boldsymbol{p}_n^{(s)} - \boldsymbol{p}_n^{(s-1)}$ to obtain a feature increment $\Delta\varphi_m^{(s)} \in \mathbb{R}^{2D_{\text{rf}}}$, then compress it by an independent Gaussian TRP matrix $P_m \sim \mathcal{N}(0,1)^{2D_{\text{rf}} \times r}$ to $u_m^{(s)} = P_m^\top \Delta\varphi_m^{(s)} \in \mathbb{R}^r$ ($r = d_{\text{feat}}$; Appendix C.4). The SORF matrices are sampled independently across levels as required by Theorem 2.

The level-m accumulator $\boldsymbol{\zeta}_m^{(s)} \in \mathbb{R}^r$, which combines the $u_m^{(s)}$ across steps via the discrete Chen recursion of RFSF-TRP [37], satisfies

$$\boldsymbol{\zeta}_1^{(s)} = u_1^{(s)}, \qquad \boldsymbol{\zeta}_m^{(s)} = \Big(\textstyle\sum_{s'<s} \boldsymbol{\zeta}_{m-1}^{(s')} \Big) \odot u_m^{(s)} \quad (m \geq 2), \tag{4}$$

where $\odot$ is the Hadamard product. Summing each level along the patch and concatenating across m yields the patch-level signature feature

$$\boldsymbol{\psi}_n = \big[c_0;\ r^{-1/2} \textstyle\sum_s \boldsymbol{\zeta}_1^{(s)};\ \ldots;\ r^{-1/2} \textstyle\sum_s \boldsymbol{\zeta}_M^{(s)} \big] \in \mathbb{R}^{D_{\mathrm{sig}}}, \quad D_{\mathrm{sig}} = 1 + Mr, \tag{5}$$

where $c_0 = 10^{-3}$ is a numerical floor. The raw features $\boldsymbol{\psi}_n$ are passed to every attention layer, which builds its own per-head, content-adaptive kernel from them (Section 4.2). Because $\boldsymbol{\psi}$ is derived from the joint path over *all* channels, it encodes cross-channel temporal structure at the path level.

4.1.2 Patch Encoder

Following the same patching scheme as Section 4.1.1 (with truncation or last-step padding so that $N \cdot P$ aligns), the patch encoder produces a per-channel representation that serves as the queries, keys, and values of attention. Observed values are retained at history positions, zero placeholders mark the query timestamps, and a binary horizon indicator $\boldsymbol{i} \in \{0,1\}^L$ flags steps belonging to the forecast horizon. Within each patch (indexed by step $s = 1, \ldots, P$), every channel is summarized by three vectors: values $\boldsymbol{x} \in \mathbb{R}^P$, timestamps $\boldsymbol{t} \in \mathbb{R}^P$, and the horizon indicator $\boldsymbol{i} \in \{0,1\}^P$. The MLP branch gates only the value channel by the observation mask $\boldsymbol{m}$, while the skip branch additionally weights all three by $\boldsymbol{m}$ before averaging only over observed positions. A two-layer MLP with GELU encodes the result into a d_{model}-dimensional patch representation:

$$\boldsymbol{h}_{c,n}^{\mathrm{mlp}} = \mathrm{MLP}\big([\boldsymbol{m}_{c,n} \odot \boldsymbol{x}_{c,n};\ \boldsymbol{t}_{c,n};\ \boldsymbol{i}_{c,n}] \big) \in \mathbb{R}^{d_{\mathrm{model}}}, \tag{6}$$

shared across patches and channels (channel-specific specialization is handled by the learnable channel embedding $\boldsymbol{e}_c$ added below). A skip connection adds the mask-weighted mean of the mask-gated triple, linearly projected to d_{model}:

$$\boldsymbol{h}_{c,n} = \boldsymbol{h}_{c,n}^{\mathrm{mlp}} + W_{\mathrm{skip}} \frac{\sum_{s=1}^{P} \boldsymbol{m}_{c,n}^{(s)} [\boldsymbol{m}_{c,n}^{(s)} \boldsymbol{x}_{c,n}^{(s)};\ \boldsymbol{t}_{c,n}^{(s)};\ \boldsymbol{i}_{c,n}^{(s)}]}{\max(\sum_s \boldsymbol{m}_{c,n}^{(s)},\ 1)}. \tag{7}$$

Horizon positions are filled with $(\boldsymbol{x}, \boldsymbol{m}) = (\boldsymbol{0}, \boldsymbol{0})$ but retain $\boldsymbol{t}$ and $\boldsymbol{i}$, so by Equation 6–Equation 7 the timestamp flows into the MLP branch while $m = 0$ excludes it from the mask-weighted skip. Adding learnable channel embeddings $\boldsymbol{e}_c \in \mathbb{R}^{d_{\mathrm{model}}}$ and positional embeddings $\boldsymbol{e}_n \in \mathbb{R}^{d_{\mathrm{model}}}$ yields the patch representation tensor $\boldsymbol{H} \in \mathbb{R}^{(B \cdot C) \times N \times d_{\mathrm{model}}}$; fully-unobserved patches are compacted before encoding, so attention operates on $N_a \leq N$ active patches.

4.2 Signature-Kernel Attention

The patch representations from the dual-path tokenization are refined through a stack of N_{layer} RUBATO blocks, each following a Pre-LN Transformer layout with residual connections throughout. Each block has three pieces: a per-head kernel built from the joint signature feature, cosine normalization, and content-dependent value gating.

Multi-head signature kernel. Each attention head $h \in \{1, \ldots, n_h\}$ (with n_h the number of heads and $d_k = d_{\mathrm{model}}/n_h$) projects the joint signature feature $\boldsymbol{\psi}_n$ from Equation 5 through its own linear map and forms the kernel by inner product. The joint signature is computed once per patch from the path covering all channels and broadcast across the per-channel batch axis:

$$\boldsymbol{\psi}_n^{(h)} = W_h^{\mathrm{sig}} \boldsymbol{\psi}_n \in \mathbb{R}^{d_k}, \qquad \bar{K}_{ij}^{(h)} = \big\langle \boldsymbol{\psi}_i^{(h)},\ \boldsymbol{\psi}_j^{(h)} \big\rangle \in \mathbb{R}^{B \times n_h \times N \times N}, \tag{8}$$

where $W_h^{\mathrm{sig}} \in \mathbb{R}^{d_k \times D_{\mathrm{sig}}}$ is learned per head *and per layer*.

Cosine normalization. The raw inner-product kernel is normalized to remove dependence on feature magnitude:

$$\widehat{K}_{ij}^{(h)} = \frac{\bar{K}_{ij}^{(h)}}{\sqrt{\bar{K}_{ii}^{(h)}\ \bar{K}_{jj}^{(h)}} + \varepsilon_{\mathrm{norm}}}, \tag{9}$$

with $\varepsilon_{\mathrm{norm}} = 10^{-8}$, placing all entries on a $[-1, 1]$ scale regardless of the norm of $\boldsymbol{\psi}$.

Content-aware value gate. A lightweight gate on the value stream lets the layer modulate how strongly each patch's value propagates, while leaving the attention distribution to the signature kernel. The hidden representation $\boldsymbol{h}_n \in \mathbb{R}^{d_{\text{model}}}$ of patch n is mapped to a per-head scalar importance score:

$$s_n^{(h)} = \sigma\big(w_h^\top \boldsymbol{h}_n\big) \in (0,1), \qquad w_h \in \mathbb{R}^{d_{\text{model}}}, \tag{10}$$

where σ is the sigmoid function. The output of the head combines kernel-based attention with the gated values:

$$\boldsymbol{A}^{(h)} = \operatorname{softmax}\big(\tau_h \cdot \widehat{\boldsymbol{K}}^{(h)}\big), \qquad \operatorname{SigAttn}(\boldsymbol{H}) = W_O \operatorname{Concat}_h\big(\boldsymbol{A}^{(h)} \left(\boldsymbol{s}^{(h)} \odot W_V^{(h)} \boldsymbol{H}\right)\big), \tag{11}$$

where $\tau_h > 0$ is a learnable per-head temperature initialized to 1 and $\boldsymbol{s}^{(h)} \in (0,1)^N$ collects the scores. The attention stage is quadratic in N; SORF affects only the projection-stage cost of the kernel feature map ($O(d_H D_{\text{rf}})$ vs. $O(D_{\text{rf}} \log d_H)$). A standard FFN sub-layer ($W_2\,\text{GELU}(W_1 \boldsymbol{z})$) and a $C \times C$ linear projection along the variate axis complete each block, all with Pre-LN normalization and residual connections.

4.3 Time-Conditioned Decoder

The decoder generates forecasts at specified future time points via per-channel cross-attention. Each target timestamp t_j^* is embedded via a learnable encoding combining a linear scale with periodic sinusoidal components: $\boldsymbol{e}_{\text{time}}(t) = [a \cdot t;\ \sin(\omega_1 t);\ \ldots;\ \sin(\omega_F t)]$. The query for channel c at time t_j^* concatenates this with the channel embedding:

$$\boldsymbol{q}_{c,j} = W_q\,[\boldsymbol{e}_{\text{time}}(t_j^*);\ \boldsymbol{e}_c] \in \mathbb{R}^{d_{\text{model}}}. \tag{12}$$

Cross-attention attends from queries to the encoded patch sequence (history-covering active patches retained after the mask-based compaction in Section 4.1), using a single linear projection W_V as both the key and value transform; a linear head then predicts scalar outputs:

$$\tilde{\boldsymbol{H}}_c = W_V\,\boldsymbol{H}_c, \qquad \boldsymbol{o}_{c,j} = \operatorname{softmax}\left(\frac{\boldsymbol{q}_{c,j}\,\tilde{\boldsymbol{H}}_c^\top}{\sqrt{d_{\text{model}}}}\right)\tilde{\boldsymbol{H}}_c, \qquad \hat{x}_{c,j} = \boldsymbol{w}_{\text{pred}}^\top \boldsymbol{o}_{c,j}. \tag{13}$$

Because queries are conditioned on continuous time embeddings, the decoder produces predictions at arbitrary, non-uniform future time points.

4.4 SORF Signature Kernel Bias Bound

The SORF projection used by the joint signature feature module introduces a structured bias relative to unbiased RFF. We bound this bias through the multi-level recursion and show it vanishes asymptotically with dimension. The bound below isolates the SORF-induced bias inside the ideal RFSF estimator [37], with TRP compression handled separately in Appendix C.4. Proofs, supporting lemmas, and the formal restatement of the assumptions on compactness, kernel bandwidth, and Hadamard-padded dimension are in Appendix C.

Lemma 1. *Under the assumptions of Appendix C, the bias function $h(z) := \mathbb{E}[\tilde{k}_{\text{SORF}}(z)] - k(z)$ satisfies, on $\mathcal{X}_\Delta \subseteq B(0,R)$,*

$$\sup_{z \in \mathcal{X}_\Delta} \|\nabla^2 h(z)\|_2 \le \frac{C_H(\alpha) \log d_H}{\sigma^2 d_H}, \tag{14}$$

with $C_H(\alpha)$ explicitly computable for any $\alpha > 2$ (Appendix C.5).

Theorem 2. *Consider the ideal recursive RFSF estimator (Equation 1 with each level using an independent SORF base feature). For the m-th order signature kernel with $\omega = V_x V_y$ (product of 1-variations):*

$$\left|\mathbb{E}\left[\tilde{K}^{(m)}\right] - K^{(m)}\right| \le \frac{[(L_k + \epsilon_{\text{bias}})^m - L_k^m]\,\omega^m}{(m!)^2}, \tag{15}$$

where $L_k = 1/\sigma^2$ and $\epsilon_{\text{bias}} = C_H(\alpha) \log d_H / (\sigma^2 d_H)$.

Corollary 3. *$\mathcal{B}^{(m)} = O(m \log d_H / d_H) \cdot L_k^{m-1} \omega^m / (m!)^2$. For the M-truncated kernel, the total bias is $O(\log d_H / d_H)$.*

Theorem 2 bounds bias only; SORF variance reduction with D_{rf} is observed empirically (Section 5) but a formal bound is open [44]. Under the canonical RFSF-TRP scaling, TRP is conditionally unbiased per signature level (Appendix C.4), so the bound transfers level-wise to the TRP-compressed kernel inner product.

5 Experiments

We evaluate RUBATO on irregular multivariate time series forecasting benchmarks (Section 5.1), followed by ablation studies (Section 5.2) and visual diagnostics (Section 5.3). Synthetic timing and bias-decay results appear in Figure 3; broader approximation-quality analyses are deferred to Appendix A.

5.1 IMTS Forecasting Benchmarks

Datasets. We evaluate on 6 irregular multivariate time series datasets spanning diverse domains, channel counts (C from 3 to 96), and missingness patterns (see Appendix B for descriptions): two clinical benchmarks (MIMIC with C=96 and PhysioNet with C=41, both $\sim$80% missing), a climate dataset (USHCN, C=5), a financial dataset (FNSPID, C=6), an air quality dataset (EPA-AIR, C=4, $\sim$70% missing), and a cloud monitoring dataset (ClusterTrace, C=3).

Baselines and protocol. We compare against 10 baselines: SeFT [11], PrimeNet [8], mTAN [35], NeuralFlows [2], t-PatchGNN [45], KAFNet [46], APN [19], Hi_Patch [20], GraFITi [42], and HyperIMTS [16]. All methods use MSE loss, 20 training epochs with early stopping patience 3, averaged over 5 seeds (mean $\pm$ std reported).

Table 1: Forecasting results on 6 irregular MTS datasets (mean$\pm$std over 5 seeds). **Bold**: best. Underline: second best. RUBATO achieves best MAE on 5/6 and best MSE on 5/6.

	MIMIC		PhysioNet		USHCN		FNSPID		EPA-AIR		ClusterTrace	
Model	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
SeFT	0.8207 ±0.0003	0.6147 ±0.0004	0.8393 ±0.0043	0.6680 ±0.0022	0.8395 ±0.0028	0.5100 ±0.0023	1.0554 ±0.1160	0.7697 ±0.0421	1.2241 ±0.2757	0.8663 ±0.1281	1.6912 ±0.0037	0.9525 ±0.0009
PrimeNet	0.8220 ±0.0001	0.6251 ±0.0001	0.8642 ±0.0000	0.6768 ±0.0000	0.9281 ±0.0002	0.5207 ±0.0006	1.1655 ±0.0021	0.8002 ±0.0003	2.6316 ±0.0052	1.4008 ±0.0016	12.6255 ±0.1164	2.6021 ±0.0140
mTAN	0.6564 ±0.0317	0.5405 ±0.0133	0.4613 ±0.0026	0.4460 ±0.0033	1.0685 ±0.1636	0.5400 ±0.0526	0.0818 ±0.0158	0.2018 ±0.0249	0.4612 ±0.0290	0.5257 ±0.0138	1.6499 ±0.1132	0.9304 ±0.0206
NeuralFlows	0.5629 ±0.0030	0.4907 ±0.0020	0.4943 ±0.0054	0.4691 ±0.0026	0.6020 ±0.0104	0.3755 ±0.0122	0.1635 ±0.0145	0.2887 ±0.0132	0.5650 ±0.0107	0.5825 ±0.0089	1.7211 ±0.0377	0.9473 ±0.0112
tPatchGNN	0.4282 ±0.0074	0.3918 ±0.0059	0.3960 ±0.0048	0.3981 ±0.0016	0.5723 ±0.0748	0.3439 ±0.0404	1.1276 ±0.0364	0.7845 ±0.0350	2.0160 ±0.7758	1.1725 ±0.3173	1.6513 ±0.0691	0.9421 ±0.0194
KAFNet	0.4426 ±0.0157	0.4091 ±0.0161	0.3927 ±0.0009	0.3987 ±0.0038	0.6660 ±0.0471	0.3962 ±0.0265	0.0941 ±0.0122	0.2060 ±0.0159	**0.4312 ±0.0044**	0.5124 ±0.0031	1.6386 ±0.0493	0.9452 ±0.0122
APN	0.4367 ±0.0080	0.3961 ±0.0071	0.3993 ±0.0031	0.3973 ±0.0046	0.6286 ±0.0469	0.3807 ±0.0262	0.0995 ±0.0154	0.2042 ±0.0142	0.4445 ±0.0057	0.5218 ±0.0047	1.4818 ±0.0307	0.9086 ±0.0045
Hi_Patch	0.4211 ±0.0125	0.3908 ±0.0109	0.3887 ±0.0041	0.3943 ±0.0028	0.6471 ±0.0054	0.3725 ±0.0139	0.0506 ±0.0065	0.1340 ±0.0095	0.4509 ±0.0173	0.5215 ±0.0137	1.4624 ±0.0264	**0.8911 ±0.0083**
GraFITi	0.3934 ±0.0091	0.3771 ±0.0064	0.3968 ±0.0133	0.3923 ±0.0055	0.5520 ±0.0075	0.3340 ±0.0051	0.1264 ±0.0037	0.2267 ±0.0023	0.5107 ±0.0190	0.5629 ±0.0127	1.5313 ±0.0466	0.9297 ±0.0057
HyperIMTS	0.3914 ±0.0145	0.3761 ±0.0075	0.3884 ±0.0023	0.3923 ±0.0054	0.5801 ±0.0655	0.3530 ±0.0421	0.1057 ±0.0214	0.1921 ±0.0212	0.5314 ±0.0148	0.5708 ±0.0119	1.5402 ±0.0661	0.9230 ±0.0102
RUBATO	**0.3780 ±0.0035**	**0.3655 ±0.0023**	**0.3811 ±0.0030**	**0.3856 ±0.0006**	**0.5133 ±0.0044**	**0.3062 ±0.0033**	**0.0402 ±0.0026**	**0.1181 ±0.0061**	0.4322 ±0.0071	**0.5097 ±0.0052**	**1.4559 ±0.1049**	0.9042 ±0.0252

Results. Table 1 compares RUBATO with 10 baselines on 6 IMTS datasets. We observe: (1) RUBATO achieves the best MAE and MSE on 5/6 datasets. The largest gain is on FNSPID (MAE 0.118 vs. 0.134 for Hi_Patch, -11.9%; -38.5% over HyperIMTS), where heterogeneous financial dynamics are captured at the path level; USHCN improves over GraFITi by $8.3\%/7.0\%$ in MAE/MSE. (2) RUBATO scales consistently: on the high-dimensional MIMIC benchmark (C=96, p=193, d_H=256) it is best on both metrics, and SORF reduces per-epoch time by 9.9% relative to dense RFF (Table 2). On EPA-AIR, KAFNet narrowly leads on MSE (0.2% gap) while RUBATO leads on MAE; on ClusterTrace, Hi_Patch edges RUBATO on MAE by 1.5% while RUBATO leads on MSE. The gains are architectural—dual-path tokenization, signature-kernel attention, and time-conditioned decoding jointly model dependencies as signature kernel similarity on the raw input, with SORF as the projection bounded by Theorem 2.

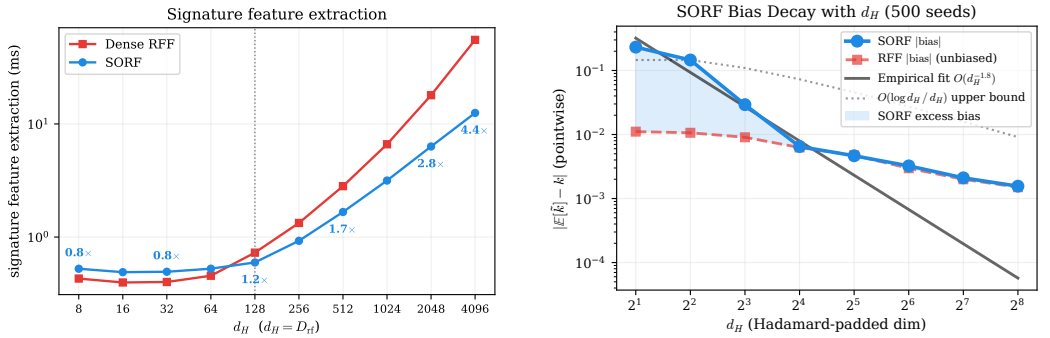
5.2 Ablation Study

We ablate four axes on USHCN (p=11, d_H=16), FNSPID (p=13, d_H=16), and MIMIC (p=193, d_H=256) with 5 seeds each (Table 2; details in Appendix A.3). Replacing signature-kernel attention by vanilla QKV attention degrades MAE on all three datasets and inflates seed-std by an order of magnitude on USHCN ($19\times$); removing the joint signature further increases error, showing that the gain comes from *using* the signature to construct the attention kernel rather than merely appending it as a feature. Signature order M=2 is the most robust default: M=3 is within seed noise on

Table 2: Ablation study on USHCN, FNSPID, and MIMIC. Numbers are mean±std over 5 seeds; relative changes w.r.t. the Signature-kernel / M=2 / per-dataset best σ_{init} / SORF baseline are shown in parentheses (best σ_{init}: 1.0 for USHCN and MIMIC, 2.0 for FNSPID, marked in bold in the σ_{init} block). Wall-clock times are per epoch; positive "SORF vs. RFF" means SORF is slower, negative means faster.

	USHCN (p=11)		**FNSPID** (p=13)		**MIMIC** (p=193)	
Variant	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**
Attention mechanism × joint-signature input						
Sig-kernel attn + joint signature (ours)	$0.3062_{\pm.003}$	$0.5133_{\pm.004}$	$0.1181_{\pm.006}$	$0.0402_{\pm.003}$	$0.3655_{\pm.002}$	$0.3780_{\pm.004}$
QKV attn + joint signature	$0.3146_{\pm.006}$ (+2.7%)	$0.5184_{\pm.005}$ (+1.0%)	$0.1194_{\pm.010}$ (+1.1%)	$0.0413_{\pm.004}$ (+2.7%)	$0.3763_{\pm.004}$ (+3.0%)	$0.3949_{\pm.011}$ (+4.5%)
QKV attn, no joint signature	$0.3404_{\pm.057}$ (+11.2%)	$0.5461_{\pm.087}$ (+6.4%)	$0.1266_{\pm.010}$ (+7.2%)	$0.0453_{\pm.004}$ (+12.7%)	$0.3736_{\pm.004}$ (+2.2%)	$0.3913_{\pm.008}$ (+3.5%)
Signature order M						
M=2 (ours)	$0.3062_{\pm.003}$	$0.5133_{\pm.004}$	$0.1181_{\pm.006}$	$0.0402_{\pm.003}$	$0.3655_{\pm.002}$	$0.3780_{\pm.004}$
M=3	$0.3068_{\pm.002}$ (+0.2%)	$0.5180_{\pm.003}$ (+0.9%)	$0.1355_{\pm.013}$ (+14.7%)	$0.0484_{\pm.005}$ (+20.4%)	$0.3713_{\pm.007}$ (+1.6%)	$0.3845_{\pm.008}$ (+1.7%)
Bandwidth σ_{init}						
σ_{init}=0.5	$0.3069_{\pm.004}$	$0.5148_{\pm.004}$	$0.1225_{\pm.005}$	$0.0436_{\pm.004}$	$0.3700_{\pm.010}$	$0.3906_{\pm.018}$
σ_{init}=1.0	$\mathbf{0.3062}_{\pm.003}$	$\mathbf{0.5133}_{\pm.004}$	$0.1228_{\pm.015}$	$0.0436_{\pm.009}$	$\mathbf{0.3655}_{\pm.002}$	$\mathbf{0.3780}_{\pm.004}$
σ_{init}=2.0	$0.3153_{\pm.013}$	$0.5193_{\pm.020}$	$\mathbf{0.1181}_{\pm.006}$	$\mathbf{0.0402}_{\pm.003}$	$0.3671_{\pm.004}$	$0.3822_{\pm.007}$
Feature projection						
SORF (ours)	$0.3062_{\pm.003}$	$0.5133_{\pm.004}$	$0.1181_{\pm.006}$	$0.0402_{\pm.003}$	$0.3655_{\pm.002}$	$0.3780_{\pm.004}$
Dense RFF	$0.3204_{\pm.023}$ (+4.6%)	$0.5254_{\pm.019}$ (+2.4%)	$0.1305_{\pm.007}$ (+10.5%)	$0.0459_{\pm.004}$ (+14.2%)	$0.3675_{\pm.004}$ (+0.5%)	$0.3836_{\pm.006}$ (+1.5%)
wall-clock time(SORF):	17.0 s		1.80 s		173 s	
wall-clock time(RFF):	15.8 s		1.60 s		192 s	
SORF vs. RFF:	+7.6%		+12.5%		−9.9%	

USHCN/MIMIC but over-parameterises FNSPID. The learnable σ_{init} is forgiving across initializations, with only FNSPID MSE showing a larger outlier. Finally, SORF matches or improves dense RFF accuracy; at d_H=16 it gives a clear MAE edge with wall-clock time within ∼13% of dense RFF, and at d_H=256 it reaches seed-noise accuracy parity while running ∼10% faster than RFF, consistent with the $O(D_{\text{rf}} \log d_H)$ vs. $O(d_H D_{\text{rf}})$ crossover (Figure 3a).



(a) Signature-extraction runtime.

(b) Pointwise SORF bias.

Figure 3: Synthetic SORF validation. (a) Full signature feature extraction with $d_H = D_{\text{rf}}$, M=2, n=16384 path steps, and TRP width r=128 on an RTX 3090. Dense RFF is faster for $d_H \leq 64$; SORF first becomes faster at the tested d_H=128 and reaches 4.43× speedup at 4096. (b) Pointwise SORF bias vs. d_H (log–log); empirical $\sim d_H^{-1.8}$ decay is faster than the $O(\log d_H / d_H)$ bound of Theorem 2 and falls below the RFF noise floor for $d_H \geq 16$.

Complexity and runtime crossover. SORF reduces the base random-feature projection from $O(d_H D_{\text{rf}})$ to $O(D_{\text{rf}} \log d_H)$; TRP, Chen recursion, and quadratic attention are unchanged. Figure 3a times the full signature extractor, including projection, differencing, TRP, and Chen recursion, on 2048 paths of length 8 with M=2, r=128, and $D_{\text{rf}} = d_H$. We report the minimum of 300 interleaved, CUDA-synchronized forward calls after 50 warm-ups, with gradients and TF32 disabled. The first sampled SORF speedup is 1.22× at d_H=128, increasing to 4.43× at 4096; dense RFF is faster for $d_H \leq 64$. For joint paths, $p = 2C + 1$ and $d_H = 2^{\lceil \log_2 p \rceil}$, so this hardware- and workload-dependent benefit applies mainly at larger channel counts, consistent with Table 2. Separately, Figure 3b shows pointwise bias over 200 pairs and 500 draws decaying as $\sim d_H^{-1.8}$; finite-feature bias–variance behavior is examined in Figure 5b.

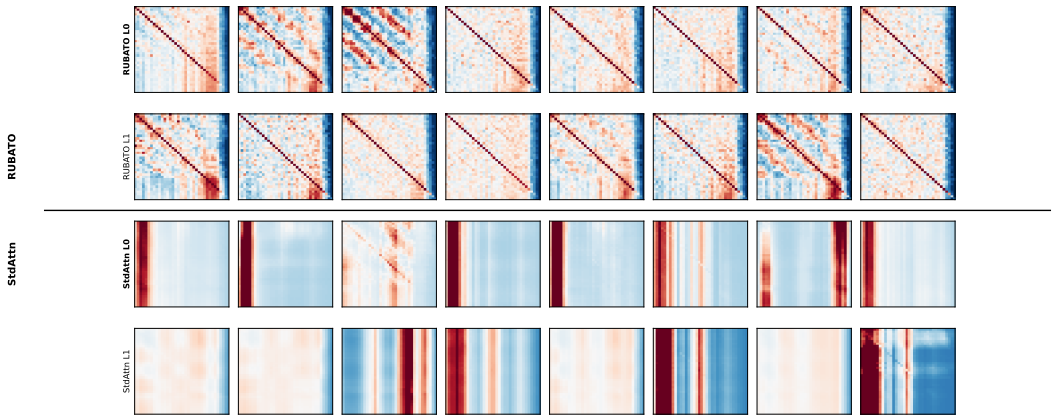


Figure 4: Per-head signature-kernel versus standard QKV attention on USHCN. RUBATO (top; two layers with eight heads each) exhibits a pronounced diagonal pattern, whereas StdAttn (bottom) shows attention collapse toward shared anchor key columns, visible as repeated vertical stripes. Red/blue indicate positive/negative deviations.

5.3 Attention Analysis

Figure 4 compares RUBATO with StdAttn, the standard QKV baseline without the joint signature in Table 2, while keeping the patch encoder and decoder unchanged. The maps visualize each head's attention deviations from $1/N$ on one USHCN test batch; the plotting protocol is given in Appendix A.4.

RUBATO exhibits a *sharp red diagonal* across heads, indicating that its strongest attention preferences follow each query's local patch position. StdAttn instead exhibits *attention collapse toward shared anchor keys*: different query rows repeatedly favour a small set of key columns, producing vertical stripes across multiple heads and layers. The preferred key positions are less query-specific.

The anchor-collapse pattern coincides with 11.2% higher MAE and 6.4% higher MSE for StdAttn on USHCN (Table 2); its MAE standard deviation across seeds also increases from 0.003 to 0.057, approximately $19\times$. Retaining the joint signature with QKV attention reduces the MAE gap to 2.7%, but still trails RUBATO. Together, the ablation and visualization support the benefit of *using* the joint signature to construct path-similarity attention, beyond merely appending it as an input feature.

6 Conclusion

We propose RUBATO, an attention framework for IMTS forecasting that combines per-channel local dynamics with the path signature of the joint multi-channel series. Both views are computed directly from raw irregular inputs, preserving time intervals, observation order, and cross-channel asynchrony without interpolation or grid alignment. Signature-kernel attention then links attention weights to similarities in the underlying path geometry. Experiments on multiple public datasets support the effectiveness of this design for IMTS forecasting.

Limitations. SORF's wall-clock benefit is workload-dependent: dense RFF is faster at small d_H, while the first measured speedup in our signature-extraction benchmark occurs at $d_H{=}128$ (Figure 3a); USHCN/FNSPID incur $7.6\%/12.5\%$ higher epoch time, whereas MIMIC is 9.9% faster (Table 2). The $O(\log d_H/d_H)$ bias bound is also loose against the empirical $\sim d_H^{-1.8}$ decay (Figure 3b), suggesting that a tighter analysis exploiting the structure of the Hadamard transform is possible. A matching variance bound, probabilistic forecasting [41, 6], and explicit missingness mechanisms remain open for future work.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (No. 62372179) and the Natural Science Basic Research Program of Shaanxi (No. 2026JC-YBMS-0710).

References

[1] Nir Ailon and Bernard Chazelle. The fast Johnson–Lindenstrauss transform and approximate nearest neighbors. *SIAM Journal on Computing*, 39(1):302–322, 2009.

[2] Marin Biloš, Johanna Sommer, Syama Sundar Rangapuram, Tim Januschowski, and Stephan Günnemann. Neural flows: Efficient alternative to neural ODEs. In *Advances in Neural Information Processing Systems*, volume 34, pages 21325–21337, 2021.

[3] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8(1):6085, 2018.

[4] Hanyin Cheng, Xingjian Wu, Xiangfei Qiu, Yang Shu, Bin Yang, and Chenjuan Guo. CCD: Capturing cross-correlations with deformable convolutional networks for multivariate time series forecasting. In *KDD*, 2026.

[5] Hanyin Cheng, Xingjian Wu, Yang Shu, Zhongwen Rao, Lujia Pan, Bin Yang, and Chenjuan Guo. CoRA: Boosting time series foundation models for multivariate forecasting through correlation-aware adapter. In *ICLR*, 2026.

[6] Hanyin Cheng, Jingrong Zhou, Yang Shu, and Chenjuan Guo. KITE: Knowledge-guided probabilistic modeling for time series forecasting with exogenous variables. In *ICML*, 2026.

[7] Ilya Chevyrev and Andrey Kormilitzin. A primer on the signature method in machine learning. In *Signature Methods in Finance: An Introduction with Computational Applications*, pages 3–64. Springer, 2025.

[8] Ranak Roy Chowdhury, Jiacheng Li, Xiyuan Zhang, Dezhi Hong, Rajesh K. Gupta, and Jingbo Shang. PrimeNet: Pre-training for irregular multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7184–7192, 2023.

[9] Edward De Brouwer, Jaak Simm, Adam Arany, and Yves Moreau. GRU-ODE-Bayes: Continuous modeling of sporadically-observed time series. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[10] Zihan Dong, Xinyu Fan, and Zhiyuan Peng. FNSPID: A comprehensive financial news dataset in time series. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

[11] Max Horn, Michael Moor, Christian Bock, Bastian Rieck, and Karsten Borgwardt. Set functions for time series. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4353–4363. PMLR, 2020.

[12] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:160035, 2016.

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

[14] Franz J. Király and Harald Oberhauser. Kernels for sequentially ordered data. *Journal of Machine Learning Research*, 20(31):1–45, 2019.

[15] Quoc Le, Tamás Sarlós, and Alexander J. Smola. Fastfood: Approximating kernel expansions in loglinear time. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 244–252. PMLR, 2013.

[16] Boyuan Li, Yicheng Luo, Zhen Liu, Junhao Zheng, Jianming Lv, and Qianli Ma. HyperIMTS: Hypergraph neural network for irregular multivariate time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 35502–35518. PMLR, 2025.

[17] Zhengyu Li, Xiangfei Qiu, Yuhan Zhu, Xingjian Wu, Jilin Hu, Chenjuan Guo, and Bin Yang. GCGNet: Graph-consistent generative network for time series forecasting with exogenous variables. In *ICLR*, 2026.

[18] Xvyuan Liu, Xiangfei Qiu, Hanyin Cheng, Xingjian Wu, Chenjuan Guo, Bin Yang, and Jilin Hu. ASTGI: Adaptive spatio-temporal graph interactions for irregular multivariate time series forecasting. In *ICLR*, 2026.

[19] Xvyuan Liu, Xiangfei Qiu, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Jilin Hu, and Bin Yang. Rethinking irregular time series forecasting: A simple yet effective baseline. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 23873–23881, 2026.

[20] Yicheng Luo, Bowen Zhang, Zhen Liu, and Qianli Ma. Hi-patch: Hierarchical patch GNN for irregular multivariate time series. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 41494–41519. PMLR, 2025.

[21] Terry J. Lyons. Differential equations driven by rough signals. *Revista Matemática Iberoamericana*, 14(2):215–310, 1998.

[22] Matthew J. Menne, Claude N. Williams, and Russell S. Vose. The United States Historical Climatology Network monthly temperature data, version 2. *Bulletin of the American Meteorological Society*, 90(7):993–1007, 2009.

[23] Fernando Moreno-Pino, Álvaro Arroyo, Harrison Waldon, Xiaowen Dong, and Álvaro Cartea. Rough transformers: Lightweight and continuous time series modelling through signature patching. In *Advances in Neural Information Processing Systems*, volume 37, pages 106264–106294, 2024. URL `https://papers.nips.cc/paper_files/paper/2024/hash/bfe167fee3be2d862a56af82dee77720-Abstract-Conference.html`.

[24] Isaac Kofi Nti, Moses Teimeh, Owusu Nyarko-Boateng, and Adebayo Felix Adekoya. Electricity load forecasting: A systematic review. *Journal of Electrical Systems and Information Technology*, 7(1):13, 2020.

[25] Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. TFB: Towards comprehensive and fair benchmarking of time series forecasting methods. In *Proc. VLDB Endow.*, pages 2363–2377, 2024.

[26] Xiangfei Qiu, Xingjian Wu, Hanyin Cheng, Xvyuan Liu, Chenjuan Guo, Jilin Hu, and Bin Yang. DBLoss: Decomposition-based loss function for time series forecasting. In *NeurIPS*, 2025.

[27] Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual clustering enhanced multivariate time series forecasting. In *SIGKDD*, pages 1185–1196, 2025.

[28] Xiangfei Qiu, Kangjia Yan, Xvyuan Liu, Xingjian Wu, and Jilin Hu. Bridging time and frequency: A joint modeling framework for irregular multivariate time series forecasting. In *ICML*, 2026.

[29] Xiangfei Qiu, Yuhan Zhu, Zhengyu Li, Xingjian Wu, Bin Yang, and Jilin Hu. DAG: A dual correlation network for time series forecasting with exogenous variables. In *ICML*, 2026.

[30] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 20, pages 1177–1184, 2007.

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

[32] Cristopher Salvi, Thomas Cass, James Foster, Terry Lyons, and Weixin Yang. The signature kernel is the solution of a Goursat PDE. *SIAM Journal on Mathematics of Data Science*, 3(3): 873–899, 2021.

[33] Mona Schirmer, Mazin Eltayeb, Stefan Lessmann, and Maja Rudolph. Modeling irregular time series with continuous recurrent units. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 19388–19405. PMLR, 2022.

[34] Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90:106181, 2020.

[35] Satya Narayan Shukla and Benjamin M. Marlin. Multi-time attention networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2021.

[36] Ikaro Silva, George Moody, Daniel J. Scott, Leo A. Celi, and Roger G. Mark. Predicting in-hospital mortality of ICU patients: The PhysioNet/Computing in Cardiology Challenge 2012. In *Computing in Cardiology*, volume 39, pages 245–248, 2012.

[37] Csaba Tóth, Harald Oberhauser, and Zoltán Szabó. Random fourier signature features. *SIAM Journal on Mathematics of Data Science*, 7(1):329–354, 2025.

[38] U.S. Environmental Protection Agency. Air Quality System (AQS). `https://www.epa.gov/air-quality`, n.d. Accessed: 2025-04-15.

[39] Qizhen Weng, Wencong Xiao, Yinghao Yu, Wei Wang, Cheng Wang, Jian He, Yong Li, Liping Zhang, Wei Lin, and Yu Ding. MLaaS in the wild: Workload analysis and scheduling in large-scale heterogeneous GPU clusters. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI '22)*, pages 945–960, 2022.

[40] Xingjian Wu, Xiangfei Qiu, Hanyin Cheng, Zhengyu Li, Jilin Hu, Chenjuan Guo, and Bin Yang. Enhancing time series forecasting through selective representation spaces: A patch perspective. In *NeurIPS*, 2025.

[41] Xingjian Wu, Xiangfei Qiu, Hongfan Gao, Jilin Hu, Bin Yang, and Chenjuan Guo. K^2VAE: A koopman-kalman enhanced variational autoencoder for probabilistic time series forecasting. In *ICML*, 2025.

[42] Vijaya Krishna Yalavarthi, Kiran Madhusudhanan, Randolf Scholz, Nourhan Ahmed, Johannes Burchert, Shayan Jawed, Stefan Born, and Lars Schmidt-Thieme. GraFITi: Graphs for forecasting irregularly sampled time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16255–16263, 2024.

[43] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 3634–3640, 2018.

[44] Felix X. Yu, Ananda Theertha Suresh, Krzysztof Choromanski, Daniel N. Holtmann-Rice, and Sanjiv Kumar. Orthogonal random features. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[45] Weijia Zhang, Chenlong Yin, Hao Liu, Xiaofang Zhou, and Hui Xiong. Irregular multivariate time series forecasting: A transformable patching graph neural networks approach. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 60179–60196. PMLR, 2024.

[46] Ziyu Zhou, Yiming Huang, Yanyun Wang, Yuankai Wu, James Kwok, and Yuxuan Liang. Revitalizing canonical pre-alignment for irregular multivariate time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 29115–29123, 2026.

A Extended Experimental Analysis

This appendix provides synthetic analyses of SORF approximation quality that support the theoretical results of Section 4.4, together with extended discussion. The projection-efficiency and bias-decay results that complement these analyses appear in the main text (Figure 3).

A.1 SORF Approximation Quality

We validate the approximation quality of SORF with two synthetic studies; the SORF/RFF error ratio across input dimensions is reported in Appendix A.2.

Approximation quality (Figure 5a). Throughout this appendix, the synthetic input dimension d refers to the Hadamard-padded block dimension d_H of the main text (powers of 2 are used directly, so no padding is needed); D abbreviates D_{rf}. We approximate the order-M=2 signature kernel on 20 variable-length paths with d_H=32, sweeping $D_{\mathrm{rf}} \in \{4, 8, \ldots, 512\}$ and averaging over 10 seeds. This diagnostic shares one random map across signature levels. Both methods' relative Frobenius errors decrease over the tested widths; SORF improves the mean error at intermediate widths (32–256), but not uniformly over D_{rf}. This finite-width trend does not establish convergence to zero at fixed d_H, since the SORF bias bound is independent of D_{rf}.

Bias–variance decomposition (Figure 5b). We expand the study to 96 configurations across six input dimensions, eight feature widths, and $M \in \{2, 3\}$. For each dimension, we generate 20 irregular paths and compare against the exact truncated signature kernel with RBF bandwidth $\sigma = 0.5$. We evaluate 190 off-diagonal Gram entries over S=50 random-feature seeds, using independent random maps at each signature level for both RFF and SORF, as in Theorem 2. This experiment isolates the kernel estimator before TRP compression. We estimate variance across seeds and correct the squared bias of the seed mean by subtracting $\widehat{\mathrm{Var}}/S$, clipping the result at zero; the plotted total is the sum of these components.

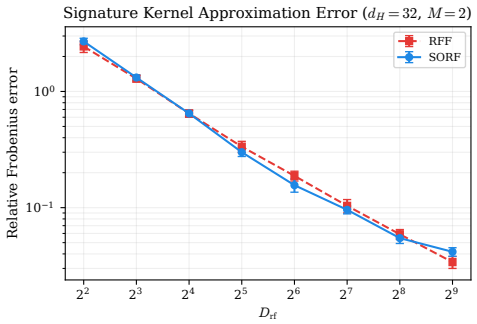
Empirical variance decreases with D_{rf} at every tested dimension for both orders. At d_H=8, SORF's squared-bias floor dominates at large widths; at d_H=16, variance reduction offsets bias at moderate widths. For $d_H \geq 32$, squared bias contributes less than 3.1% of SORF's total error, and SORF MSE is comparable to or below RFF (maximum SORF/RFF ratio ≈ 1.032). At d_H=32 and $D_{\mathrm{rf}} \in \{32, 64, 128, 256\}$, SORF reduces total MSE by approximately 21–23%. For $d_H \geq 32$, matched M=2 and M=3 MSE ratios differ by less than 0.015, indicating similar relative behavior across these orders. These observations provide empirical evidence of controlled variance in the tested regime; a formal SORF variance bound remains open [44].

A.2 SORF/RFF Error Ratio Across Input Dimensions

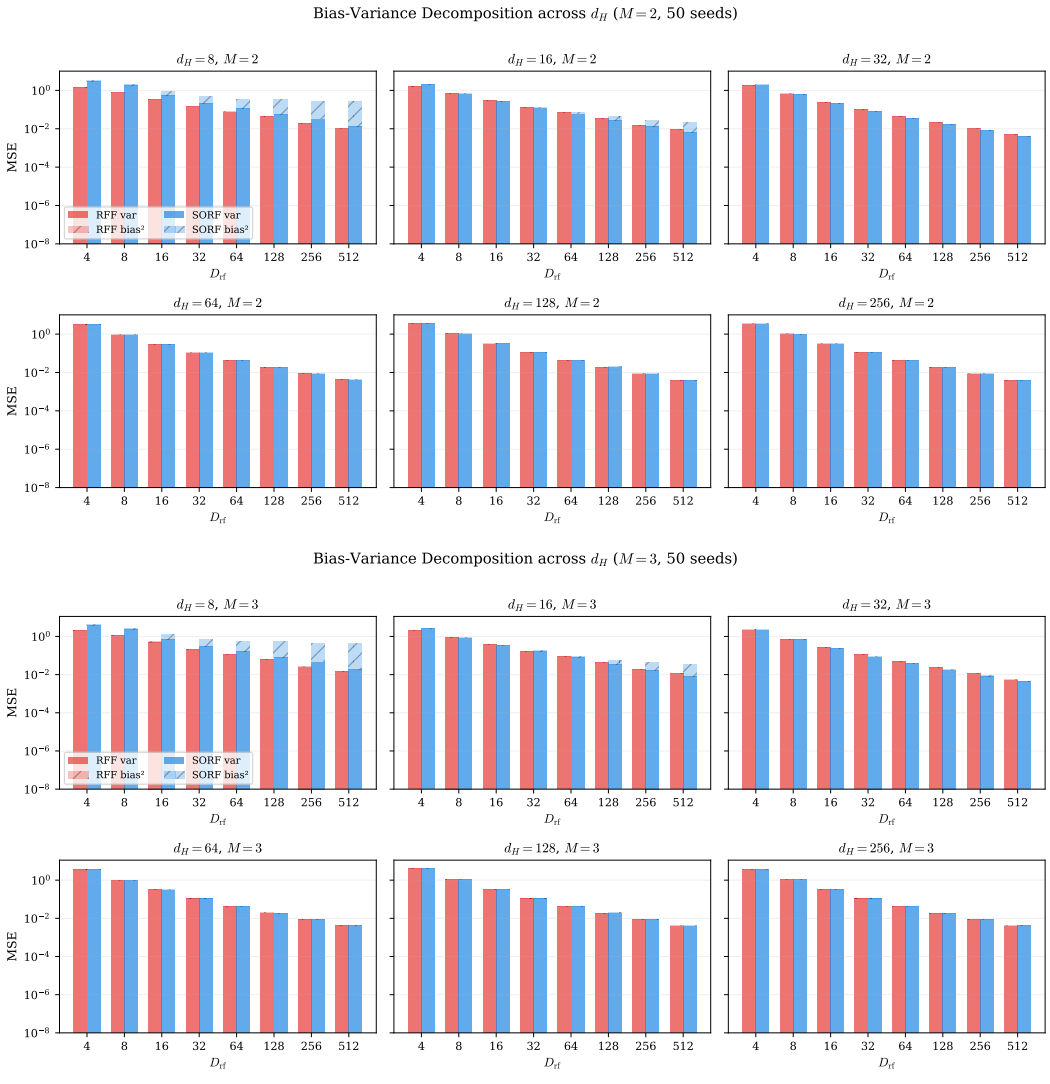
For each Hadamard-padded dimension $d_H \in \{16, 32, 64, 128, 256\}$ we compute the ratio of mean SORF error to mean RFF error on the order-2 signature kernel, sweeping $D_{\mathrm{rf}} \in \{4, \ldots, 512\}$ and averaging over 10 independent seeds. The ratio sits consistently below 1.0 across most (d_H, D_{rf}) configurations, with the best ratio of 0.83 observed at $d_H = 32$, and the advantage is stable across dimensions. This pattern is empirical evidence that, at the (d_H, D_{rf}) configurations tested, the $O(\log d_H / d_H)$ SORF bias (Lemma 1) is compensated by an observed variance reduction relative to dense RFF (consistent with the RBF-kernel pattern reported by Yu et al. [44]), and as d_H grows the bias term decays (Theorem 2), explaining the cross-dimensional stability. At large D_{rf} the ratio can exceed 1.0: once D_{rf} is large enough that variance becomes negligible, SORF's irreducible bias (which does not decay with D_{rf}) becomes the dominant error component, exactly as predicted by Theorem 2. The broader independent-level study in Figure 5b shows that this effect is already visible within the tested widths at $d_H \in \{8, 16\}$.

A.3 Ablation Study Details

Table 2 (in the main text, Section 5.2) reports the ablation study on three IMTS benchmarks that together span the relevant joint-path dimension p=2C+1: USHCN (short-sequence, low-p climate data; C=5, p=11, L=24, P=6), FNSPID (long-sequence, low-p financial data; C=6, p=13, L=730, P=30), and MIMIC (short-sequence, high-p clinical data; C=96, p=193, L=24, P=6).



(a) Approx. error vs. D_{rf}.



(b) Bias–variance decomposition: M=2 (top) and M=3 (bottom).

Figure 5: SORF vs. RFF for signature kernel approximation. (a) Relative Frobenius error of a shared-map kernel approximation on 20 paths at d_H=32, M=2 (10 seeds). (b) Stacked empirical variance (solid) and squared bias (hatched, corrected for finite-seed estimation) over 50 seeds, with independent random maps across signature levels. The 96 configurations span $d_H \in \{8, 16, 32, 64, 128, 256\}$, $D_{\mathrm{rf}} \in \{4, 8, \ldots, 512\}$, and $M \in \{2, 3\}$; the vertical axes are logarithmic.

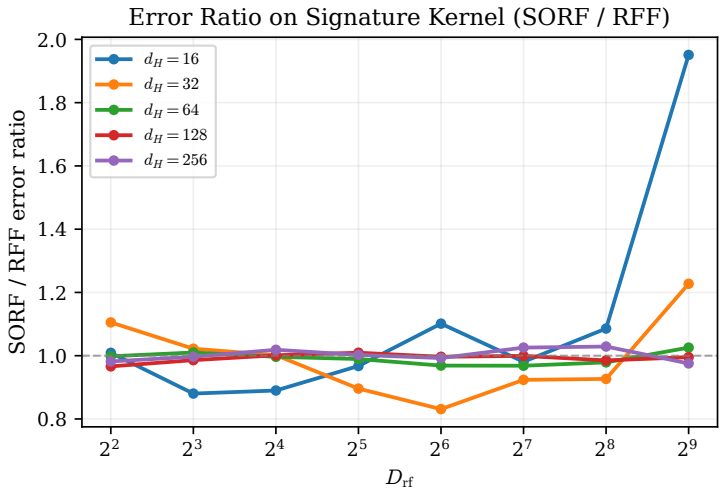


Figure 6: Ratio of SORF to RFF signature kernel approximation error across Hadamard-padded dimensions $d_H \in \{16, 32, 64, 128, 256\}$ and random feature dimensions $D_{\text{rf}} \in \{4, \ldots, 512\}$. Ratios below the dashed line indicate SORF outperforms RFF.

All three columns use the same fixed configuration as the main forecasting table (Table 1): the "Signature-kernel attention + joint signature / M=2 / per-dataset best σ_{init} / SORF" row reproduces the headline RUBATO numbers within 10^{-4} (per-dataset best σ_{init}: 1.0 for USHCN and MIMIC, 2.0 for FNSPID). All rows: 5 seeds (base 2022), 20 epochs, patience 3, MSE loss.

Attention mechanism vs. joint signature input. The first block of Table 2 disentangles the attention mechanism from the input feature with a 3-way comparison: (a) sig-kernel attention conditioned on the joint-path signature (RUBATO); (b) vanilla `nn.MultiheadAttention` that still receives the joint signature, projected to d_{model} and added as a per-patch bias to the patch representation tensor ("QKV + joint sig"); and (c) vanilla MHA with the joint signature *dropped entirely* ("QKV, no sig"). Variant (a) beats (c) on every dataset. The MAE gap is most pronounced on USHCN ($+11.2\%$), where (c)'s seed std also blows up from 0.003 to 0.057, coinciding with the anchor-collapse pattern in Figure 4; FNSPID and MIMIC show smaller but consistent degradations ($+7.2\%$ and $+2.2\%$ MAE). Feeding the joint signature into vanilla QKV (variant (b)) recovers most of this gap on USHCN and FNSPID but leaves a $+3.0\%$ MAE residual on MIMIC, and (b) trails (a) on every dataset. The gain therefore comes from *using* the signature to construct a path-similarity kernel, not from *having access* to it as a feature.

This conclusion concerns *how* the signature is used, not *which order* to use; the choice of truncation level M matters separately, and we discuss it next.

Signature order M. We omit M=1 because first-order iterated integrals reduce to per-channel path increments $\int dX^i = X_T^i - X_0^i$ with no cross-channel term, so the M=1 signature cannot encode the multi-variable interactions that the signature-kernel attention is designed to exploit. Comparing the meaningful choices M=2 vs. M=3: on USHCN and MIMIC the two are within one seed-standard-deviation, but M=3 over-parameterises FNSPID's 6-channel joint path ($+14.7\%$ MAE, $+20.4\%$ MSE). M=2 is the lowest-regret default without dataset-specific tuning; higher M marginally helps high-channel datasets such as MIMIC.

Bandwidth σ_{init}. Although the learnable $\log \sigma$ absorbs most of the initialisation, the scan over $\sigma_{\text{init}} \in \{0.5, 1.0, 2.0\}$ reveals a mild dataset preference: USHCN and MIMIC favour σ=1.0, FNSPID favours σ=2.0. The worst single choice stays within $\sim$4% of the per-dataset best on MAE across all three datasets, and FNSPID MSE is the only outlier at $+8.5\%$. A single safe default is adequate without tuning; per-dataset selection (used in Table 1) yields modest gains.

SORF vs. dense RFF. At low d_H the structured orthogonality of SORF reduces variance enough to give a measurable accuracy edge over dense RFF: on USHCN (p=11→d_H=16) SORF improves MAE/MSE by $4.6\%/2.4\%$, and on FNSPID (p=13→d_H=16) by $10.5\%/14.2\%$, consistent with the sub-1.0 SORF/RFF kernel-error ratios in Figure 6. At p=193→d_H=256 (MIMIC) the two methods sit within seed noise (RFF $+0.5\%$ MAE, $+1.5\%$ MSE), supporting comparable forecasting accuracy at the tested feature widths. Wall-clock behavior reflects the projection-cost difference, $O(d_H D_{\mathrm{rf}})$ vs. $O(D_{\mathrm{rf}} \log d_H)$. At small d_H the Walsh–Hadamard launch cost still trails GPU GEMM on tiny inputs, so SORF runs 7.6% slower per epoch than dense RFF on USHCN and 12.5% slower on FNSPID; once d_H=256 on MIMIC, SORF is 9.9% faster, and the full signature-extraction benchmark (Figure 3a) first favors SORF at the sampled d_H=128 ($1.22\times$) and reaches $4.43\times$ speedup at d_H=4096. SORF therefore matches dense RFF in accuracy at every d_H we tested, with a clear edge at low d_H, and the projection-stage efficiency advantage engages as d_H grows.

A.4 Additional Visualization Details

Section 5.3 presents the USHCN attention comparison (Figure 4). Here we provide its plotting protocol and a complementary encoder-representation diagnostic on MIMIC. Both use one test batch passed through trained checkpoints. StdAttn is the "QKV attn, no joint signature" ablation in Table 2; the patch encoder and decoder are unchanged.

Attention visualization. Key padding masks are applied before softmax. RUBATO shares attention across channels, whereas StdAttn computes channel-specific weights; post-softmax weights are averaged over samples and channels for each head. The saved maps retain the batch-padded token grid, with $N = 36$, and we subtract the nominal uniform reference $1/N$. Each model and layer uses a separate symmetric colour scale clipped at the 95th percentile of absolute deviations across its heads. Colours therefore indicate relative patterns, and their magnitudes should not be compared across models or layers.

Encoder-token PC1. PCA is fitted separately to each model's active encoder tokens for the selected channel. The horizontal coordinate is the normalized compact patch index $n/(N_a - 1)$, where N_a is the number of active patches, rather than elapsed physical time. The encoded sequence includes the observed history and future query positions. The sign of PC1 is chosen independently for each model to make its Spearman correlation with patch index nonnegative, so comparisons concern the absolute correlation.

Figure 7 shows the analysis on MIMIC channel 17 (213 active tokens). The absolute correlations are 0.752 and 0.396, respectively. Thus, in this displayed case, the leading direction of encoder variation is more strongly associated with patch order in RUBATO. This visualization describes a selected batch and channel; it does not establish that the same pattern holds across all channels or that PC1 captures a physical seasonal process.

B Datasets

This section describes the six irregular multivariate time series (IMTS) benchmarks used in our experiments.

MIMIC-III [12] is a freely accessible ICU database covering over 50,000 adult admissions (2001–2012). Following t-PatchGNN [45] and HyperIMTS [16], we extract per-stay trajectories over C=96 clinical channels (vital signs and lab tests); the $\sim$80% missingness reflects routine asynchronous bedside-and-lab sampling.

PhysioNet [36] was released for the 2012 PhysioNet/CinC Challenge on in-hospital mortality. It contains 12,000 ICU admissions with up to 41 time-series clinical variables over the first 48 hours of each stay. Observations are at irregular bedside-driven intervals ($\sim$80% missing); we adopt the standard patient-level split used by prior IMTS benchmarks.

USHCN [22] is the U.S. Historical Climatology Network monthly dataset used by NOAA. It contains 1,218 weather stations with C=5 daily meteorological variables (precipitation, snowfall, snow depth, min/max temperature). We follow the standard IMTS sliding-window protocol with station-level stratified splits; missingness is moderate and arises from station reporting gaps.

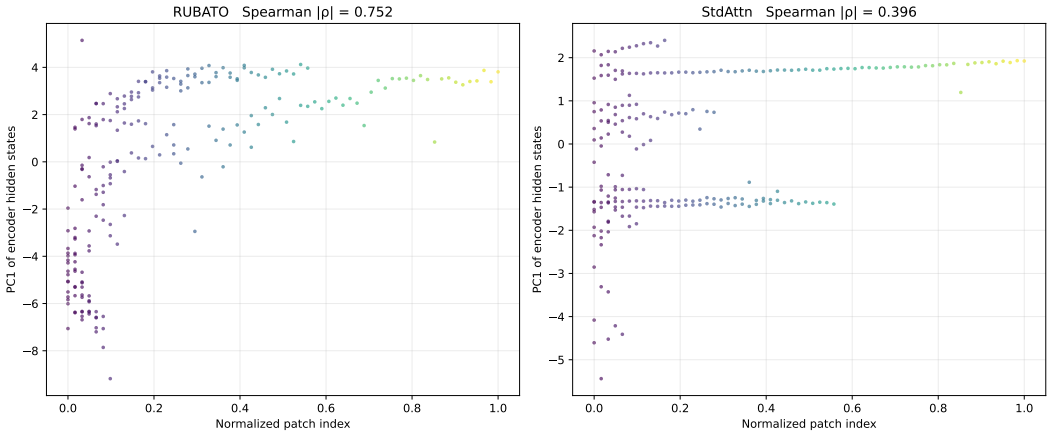


Figure 7: Encoder PC1 versus normalized compact patch index on MIMIC channel 17, using all 213 active tokens from one test batch. PCA is fitted and its sign aligned separately for each model; $|\rho| = 0.752$ for RUBATO (left) and 0.396 for StdAttn (right). Colours encode patch order.

FNSPID [10] is the Financial News and Stock Price Integration Dataset on S&P-500 constituents (1999–2023). We retain the numerical price stream and curate a 10-ticker, C=6 panel (open, high, low, close, adjusted close, volume). Irregularity is structural: differing listing windows (∼3,000–3,700 days) yield asymmetric missingness once globally aligned.

EPA-AIR [38] is derived from the U.S. EPA Air Quality System. We construct an hourly irregular series with C=4 channels (temperature, PM2.5, AQI, ozone) from a representative set of monitoring stations. The ∼70% missingness reflects asynchronous calibration cycles and per-station reporting policies; we use a longer history/horizon (336/168) than the clinical settings.

ClusterTrace is built from the Alibaba cluster-trace-gpu-v2020 release [39], a production trace of ∼6,500 GPUs across ∼1,800 machines (July–August 2020). We aggregate it into a C=3 series of active instances, tasks, and jobs per window. Missingness is low, but the dynamics are bursty and heavy-tailed across all three scheduling levels.

C Full Proofs

This appendix provides complete proofs for all theoretical results stated in Section 4.4. We begin with supporting lemmas (Section C.1), then give the full proof of the Hessian bound (Lemma 1) in Section C.2, the signature kernel bias bound (Theorem 2) in Section C.3, and the TRP extension in Section C.4. Numerical verification of constants appears in Section C.5.

Assumptions. We assume throughout that (a) the SORF input domain $\mathcal{X} \subset \mathbb{R}^{d_H}$ is compact convex with diameter R, where d_H is the Hadamard-padded dimension (Section 3); (b) the base kernel is Gaussian with bandwidth $\sigma \geq \sigma_{\min} > 0$ (preventing L_k blow-up if σ is learnable); (c) d_H is a power of 2 by construction; and (d) $d_H \geq d_0(R, \sigma)$ with $d_0/\log d_0 \geq c \max\{(R/\sigma)^2, (R/\sigma)^4\}$.

C.1 Supporting Lemmas

Lemma 4. *Let $k(u, v) = \exp(-\|u - v\|^2/(2\sigma^2))$ be the Gaussian kernel and define the cross-difference operator $\delta^2_{i,j} k := k(x_i, y_j) - k(x_i, y_{j-1}) - k(x_{i-1}, y_j) + k(x_{i-1}, y_{j-1})$. Then*

$$|\delta^2_{i,j} k| \leq L_k \, \delta\omega_{i,j}, \tag{16}$$

where $L_k = 1/\sigma^2$ and $\delta\omega_{i,j} = \|x_i - x_{i-1}\| \, \|y_j - y_{j-1}\|$.

Proof. Write $\delta^2_{i,j} k$ as a double integral via the integral mean value theorem:

$$\delta^2_{i,j} k = \int_0^1 \int_0^1 (\Delta x_i)^\top \nabla^2 f(a + s\,\Delta x_i - t\,\Delta y_j)\,(-\Delta y_j)\, ds\, dt, \tag{17}$$

where $a = x_{i-1} - y_{j-1}$, $\Delta x_i = x_i - x_{i-1}$, $\Delta y_j = y_j - y_{j-1}$, and $f(z) = \exp(-\|z\|^2/(2\sigma^2))$. The Hessian is

$$\nabla^2 f(z) = f(z)\left(\frac{zz^\top}{\sigma^4} - \frac{I}{\sigma^2}\right).$$

The matrix $zz^\top/\sigma^4 - I/\sigma^2$ has eigenvalue $\|z\|^2/\sigma^4 - 1/\sigma^2$ (along z, multiplicity 1) and $-1/\sigma^2$ (orthogonal directions, multiplicity $d-1$). Writing $u = \|z\|^2/\sigma^2$, the spectral norm of $\nabla^2 f(z)$ is $\frac{f(z)}{\sigma^2}\max(u-1, 1)$. For $0 \le u \le 2$: $f(z) \cdot 1/\sigma^2 \le 1/\sigma^2$ (attained at $u = 0$). For $u > 2$: $f(z)(u-1)/\sigma^2 = (u-1)e^{-u/2}/\sigma^2$, maximized at $u = 3$ giving $2e^{-3/2}/\sigma^2 \approx 0.446/\sigma^2 < 1/\sigma^2$. Therefore $\sup_z \|\nabla^2 f(z)\|_2 = 1/\sigma^2 =: L_k$, attained at $z = 0$. The bound follows:

$$|\delta^2_{i,j} k| \le L_k \|\Delta x_i\| \, \|\Delta y_j\| = L_k \, \delta\omega_{i,j}. \qquad \square$$

Lemma 5. *Define $r(x) := \ln\cos x + x^2/2$. For $|x| \le \pi/4$:*

$$|r(x)| \le C_r \, x^4, \qquad |r'(x)| \le C_r \, |x|^3, \qquad |r''(x)| \le C_r \, x^2, \tag{18}$$

where $C_r = (4/\pi)^2 \approx 1.62$. Furthermore, $r(x) \le 0$ for $|x| < \pi/2$.

Proof. Direct computation gives $r''(x) = -\tan^2 x$. On $[0, \pi/4]$, the function $\tan x/x$ is increasing (since $(\tan x/x)' = (x\sec^2 x - \tan x)/x^2 > 0$ for $x \in (0, \pi/2)$), so

$$\sup_{|x| \le \pi/4} \frac{|\tan x|}{|x|} = \frac{\tan(\pi/4)}{\pi/4} = \frac{4}{\pi},$$

giving $|r''(x)| = \tan^2 x \le (4/\pi)^2 x^2 = C_r \, x^2$. Since $r'(x) = -\tan x + x$ and $r'(0) = 0$, integrating this second-derivative bound yields

$$|r'(x)| = \left|\int_0^x r''(t)\, dt\right| \le \int_0^{|x|} C_r \, t^2 \, dt = \frac{C_r}{3} \, |x|^3 \le C_r \, |x|^3,$$

and integrating once more, using $r(0) = 0$ and $r'(0) = 0$,

$$|r(x)| = \left|\int_0^x r'(t)\, dt\right| \le \int_0^{|x|} C_r \, t^3 \, dt = \frac{C_r}{4} \, x^4 \le C_r \, x^4.$$

For non-positivity, note that $r''(x) = -\tan^2 x \le 0$ for $|x| < \pi/2$, so r' is decreasing on $[0, \pi/2)$ with $r'(0) = 0$ and hence $r'(x) \le 0$ for $x > 0$; by symmetry $r'(x) \ge 0$ for $x < 0$, so r attains its maximum at $x = 0$ where $r(0) = 0$, giving $r(x) \le 0$. $\square$

Lemma 6. *Let $\omega_{i,j} = \sum_{i' \le i} \sum_{j' \le j} \delta\omega_{i',j'}$ denote the cumulative product variation up to index (i, j), with $\delta\omega_{i',j'} = \|\Delta x_{i'}\| \, \|\Delta y_{j'}\|$. Write $\omega_{S,T} = A \cdot B$ where $A = \sum_{i=1}^S a_i$, $B = \sum_{j=1}^T b_j$ with $a_i = \|\Delta x_i\|$, $b_j = \|\Delta y_j\|$, and $A_i = \sum_{i'=1}^i a_{i'}$, $B_j = \sum_{j'=1}^j b_{j'}$. Then for any $m \ge 1$:*

$$\sum_{i=1}^S \sum_{j=1}^T \frac{\omega_{i-1,j-1}^{m-1}}{((m-1)!)^2} \, \delta\omega_{i,j} \le \frac{\omega_{S,T}^m}{(m!)^2}. \tag{19}$$

Proof. Using the product structure $\omega_{i-1,j-1} = A_{i-1} B_{j-1}$, the left-hand side factors as

$$\left(\sum_{i=1}^S \frac{A_{i-1}^{m-1}}{(m-1)!} \, a_i\right)\left(\sum_{j=1}^T \frac{B_{j-1}^{m-1}}{(m-1)!} \, b_j\right).$$

It suffices to show that each factor is bounded by the corresponding variation raised to the m-th power divided by $m!$. Define $F_i = A_i^m/m!$. By convexity of $t \mapsto t^m$, the mean value theorem gives $A_i^m - A_{i-1}^m \ge m \, A_{i-1}^{m-1}(A_i - A_{i-1}) = m \, A_{i-1}^{m-1} a_i$, so $\Delta F_i = (A_i^m - A_{i-1}^m)/m! \ge A_{i-1}^{m-1} a_i/(m-1)!$. Summing:

$$\sum_{i=1}^S \frac{A_{i-1}^{m-1}}{(m-1)!} \, a_i \le \sum_{i=1}^S \Delta F_i = F_S = \frac{A_S^m}{m!}.$$

The same bound applies to the B sum. Multiplying:

$$\text{LHS of Equation 19} \le \frac{A_S^m}{m!} \cdot \frac{B_T^m}{m!} = \frac{\omega_{S,T}^m}{(m!)^2}. \qquad \square$$

C.2 Proof of Lemma 1

We now give the complete proof of the SORF bias function Hessian bound. Recall that $h(z) = \mathbb{E}[\tilde{k}_{\mathrm{SORF}}(z)] - k(z)$ where $k(z) = e^{-\|z\|^2/(2\sigma^2)}$ (abusing notation to write $k(z) = k(u, v)$ for $z = u - v$), and the SORF random feature matrix is $W_{\mathrm{SORF}} = \frac{\sqrt{d_H}}{\sigma} HD_1HD_2HD_3 \in \mathbb{R}^{d_H \times d_H}$, with H the normalized Hadamard matrix and D_1, D_2, D_3 independent diagonal Rademacher matrices. *Notation.* For readability we write $d \equiv d_H$ throughout this appendix proof; every occurrence of d below refers to the Hadamard-padded SORF block dimension.

The argument has five phases: a product factorization for $\mathbb{E}[\tilde{k}(z)]$, a log-correction decomposition isolating the bias function $S(z)$, a high-probability dispersion bound on the Hadamard projections, a good-event Hessian bound, and a crude bound on the bad event.

We begin with the product factorization. The SORF kernel is $\tilde{k}(z) = \frac{1}{d}\sum_{i=1}^{d} \cos(w_i^\top z)$, so $\mathbb{E}[\tilde{k}(z)] = \frac{1}{d}\sum_i \mathbb{E}[\cos(w_i^\top z)]$. Define $v = HD_3z$ and $\beta_k = (HD_2v)_k$. The first row gives $w_1^\top z = \frac{\sqrt{d}}{\sigma}(HD_1HD_2HD_3z)_1 = \frac{1}{\sigma}\sum_{k=1}^{d} \epsilon_k\beta_k$, where $\epsilon_k = (D_1)_{kk}$ are independent Rademacher variables. Since $\mathrm{sign}((H)_{ik})\epsilon_k$ has the same distribution as ϵ_k (both are Rademacher) and they remain mutually independent, all rows have the same expected cosine: $\mathbb{E}[\cos(w_i^\top z)] = \mathbb{E}[\cos(w_1^\top z)]$ for every i. Therefore $\mathbb{E}[\tilde{k}(z)] = \mathbb{E}[\cos(w_1^\top z)]$. Conditioning on D_2, D_3 and using the independence of the ϵ_k:

$$\Phi(z) := \mathbb{E}_{D_1}\left[\cos(w_1^\top z)\right] = \mathbb{E}_{D_1}\left[\cos\left(\frac{1}{\sigma}\sum_{k=1}^{d}\epsilon_k\beta_k\right)\right] = \prod_{k=1}^{d}\cos(\beta_k/\sigma), \tag{20}$$

where the last equality follows from the identity $\mathbb{E}[\cos(\sum_k \epsilon_k\alpha_k)] = \prod_k \mathbb{E}[\cos(\epsilon_k\alpha_k)] = \prod_k \cos(\alpha_k)$ for independent Rademacher ϵ_k (since $\mathbb{E}[\cos(\epsilon\alpha)] = \cos(\alpha)$ for Rademacher ϵ). Thus $\mathbb{E}[\tilde{k}(z)] = \mathbb{E}_{D_2,D_3}[\Phi(z)]$ and $h(z) = \mathbb{E}_{D_2,D_3}[\Phi(z)] - g(z)$ where $g(z) = e^{-\|z\|^2/(2\sigma^2)}$.

We next decompose Φ around the target Gaussian. On the event that $|\beta_k/\sigma| < \pi/2$ for all k (which the dispersion estimate below shows holds with high probability), each $\cos(\beta_k/\sigma) > 0$, and

$$\ln\Phi(z) = \sum_{k=1}^{d}\ln\cos(\beta_k/\sigma) = \sum_{k=1}^{d}\left[-\frac{\beta_k^2}{2\sigma^2} + r(\beta_k/\sigma)\right] = -\frac{\|z\|^2}{2\sigma^2} + S(z),$$

where $S(z) = \sum_{k=1}^{d} r(\beta_k/\sigma)$ and the identity $\sum_k \beta_k^2 = \|z\|^2$ follows from the orthogonal invariance of the Hadamard transform (H is orthogonal, so $\sum_k \beta_k^2 = \|HD_2v\|^2 = \|D_2v\|^2 = \|v\|^2 = \|HD_3z\|^2 = \|z\|^2$). Therefore

$$\Phi(z) = g(z)\, e^{S(z)}, \qquad \Phi(z) - g(z) = g(z)\big(e^{S(z)} - 1\big). \tag{21}$$

Since $r(x) \le 0$ for $|x| < \pi/2$ by Lemma 5, $S(z) = \sum_k r(\beta_k/\sigma) \le 0$ on the good event, so $Q := e^S \le 1$ and the unconditional bound

$$|e^S - 1| = 1 - e^S \le -S = |S| \tag{22}$$

follows from $e^t \ge 1 + t$ applied to $t = S \le 0$, requiring no further condition on $|S|$.

We now control the Hadamard projections. Fix z with $\|z\| \le R$ (zero-padded into $\mathbb{R}^{d_H}$ if necessary). For each coordinate k, $\beta_k = \sum_\ell H_{k\ell}\,\eta_\ell\, v_\ell$, where $\eta_\ell = (D_2)_{\ell\ell}$ are independent Rademacher and $v_\ell = (HD_3z)_\ell$ are fixed given D_3. The summands $H_{k\ell}\eta_\ell v_\ell$ are independent and mean-zero. With $|H_{k\ell}| = 1/\sqrt{d_H}$, the per-coordinate weights satisfy $\sum_\ell H_{k\ell}^2 v_\ell^2 = \frac{1}{d_H}\sum_\ell v_\ell^2 = \|v\|^2/d_H = \|z\|^2/d_H$ (using $\|v\| = \|HD_3z\| = \|z\|$). Hoeffding's inequality (in its $\sum c_\ell^2$ form for bounded summands $c_\ell\eta_\ell$) therefore gives, for $z \neq 0$,

$$\Pr[|\beta_k| > t] \;\le\; 2\exp\!\left(-\frac{t^2}{2\sum_\ell H_{k\ell}^2 v_\ell^2}\right) \;=\; 2\exp\!\left(-\frac{d_H\, t^2}{2\|z\|^2}\right);$$

the case $z = 0$ is trivial since $\beta_k = 0$ deterministically. Taking $t = C_0\|z\|\sqrt{\log d_H/d_H}$ with $C_0 = \sqrt{2(\alpha+1)}$ and applying a union bound over $k = 1, \ldots, d_H$:

$$\Pr[\mathcal{G}(z)] := \Pr\left[\max_{1\le k\le d_H}|\beta_k| \le C_0\|z\|\sqrt{\frac{\log d_H}{d_H}}\right] \ge 1 - 2d_H^{-\alpha}. \tag{23}$$

On the good event $\mathcal{G}(z)$, we have $\eta := \max_k |\beta_k| \le C_0\|z\|\sqrt{\log d/d}$. For $d \ge d_0$ (where d_0 ensures $C_0(R/\sigma)\sqrt{\log d/d} \le \pi/4$), Lemma 5 applies. Let $a_k \in \mathbb{R}^d$ denote the k-th column of $(HD_2HD_3)^\top$, so $\beta_k = a_k^\top z$. Since HD_2HD_3 is orthogonal, $\{a_k\}_{k=1}^d$ is an orthonormal basis, giving $\nabla\beta_k = a_k$ and

$$\nabla S = \frac{1}{\sigma}\sum_k r'(\beta_k/\sigma)\, a_k, \qquad \nabla^2 S = \frac{1}{\sigma^2}\sum_k r''(\beta_k/\sigma)\, a_k a_k^\top. \tag{24}$$

We first bound $|S|$, $\|\nabla S\|$, $\|\nabla^2 S\|_2$, and then assemble them through the product rule. Using $|r(x)| \le C_r x^4$ and $\sum_k \beta_k^4 \le \eta^2 \sum_k \beta_k^2 = \eta^2\|z\|^2$,

$$|S(z)| \le \frac{C_r}{\sigma^4}\sum_k \beta_k^4 \le \frac{C_r\eta^2\|z\|^2}{\sigma^4} \le \frac{C_r C_0^2 R^4 \log d}{\sigma^4 d},$$

and Equation 22, $S \le 0$, gives $|e^S - 1| \le |S|$ unconditionally. For the gradient, orthonormality of $\{a_k\}$ gives $\|\sum_k c_k a_k\|^2 = \sum_k c_k^2$, so $|r'(x)| \le C_r|x|^3$ and $\sum_k |\beta_k|^6 \le \eta^4\|z\|^2$ yield

$$\|\nabla S\|^2 = \frac{1}{\sigma^2}\sum_k |r'(\beta_k/\sigma)|^2 \le \frac{C_r^2\eta^4\|z\|^2}{\sigma^8}, \quad \text{so} \quad \|\nabla S\| \le \frac{C_r C_0^2\|z\|^3\log d}{\sigma^4 d}.$$

For the Hessian, $\{a_k\}$ orthonormal also gives $\|\sum_k \lambda_k a_k a_k^\top\|_2 = \max_k|\lambda_k|$, and with $|r''(x)| \le C_r x^2$,

$$\|\nabla^2 S\|_2 = \frac{1}{\sigma^2}\max_k |r''(\beta_k/\sigma)| \le \frac{C_r\eta^2}{\sigma^4} \le \frac{C_r C_0^2\|z\|^2\log d}{\sigma^4 d}.$$

We now apply the product rule to $\nabla^2(\Phi - g) = \nabla^2(g(Q-1))$ where $Q = e^S$:

$$\nabla^2(g(Q{-}1)) = (Q{-}1)\nabla^2 g + 2(\nabla g)(\nabla(Q{-}1))_{\mathrm{sym}}^\top + g\,\nabla^2(Q{-}1). \tag{25}$$

Write $u = \|z\|/\sigma$ and use $e^S \le 1$ (since $S \le 0$). The dominant term comes from $g\,\nabla^2(Q-1)$. With $\nabla^2 Q = e^S(\nabla^2 S + \nabla S\,\nabla S^\top)$ and $e^S \le 1$, the Hessian-of-S bound above gives

$$g\|\nabla^2 Q\|_2 \le g\big(\|\nabla^2 S\|_2 + \|\nabla S\|^2\big) \le \frac{C_r C_0^2 g\, u^2 \log d}{\sigma^2 d} + \underbrace{g\|\nabla S\|^2}_{=\mathcal{R}(z)},$$

where $\mathcal{R}(z) \le C_r^2 C_0^4 g\, u^6 \log^2 d/(\sigma^2 d^2)$ is $O(\log^2 d/d^2)$, strictly lower order than the $O(\log d/d)$ main term and absorbed below. The cross term $2(\nabla g)(\nabla(Q-1))_{\mathrm{sym}}^\top$ is bounded by combining $\|\nabla(Q{-}1)\| = e^S\|\nabla S\| \le \|\nabla S\| \le C_r C_0^2 u^3 \log d/(\sigma d)$ with $\|\nabla g\| = gu/\sigma$, giving a contribution at most $2C_r C_0^2 g\, u^4 \log d/(\sigma^2 d)$. The first term $(Q-1)\nabla^2 g$ uses the *pointwise* Hessian bound $\|\nabla^2 g(z)\|_2 = \frac{g(z)}{\sigma^2}\max(u^2-1,\,1) \le \frac{g(z)(1+u^2)}{\sigma^2}$ (not the global $1/\sigma^2$, since it is multiplied by the z-dependent factor $|Q-1|$); together with $|Q-1| \le |S| \le C_r C_0^2 u^4 \log d/d$ from Equation 22, this contributes at most $C_r C_0^2 g(z)(u^4+u^6)\log d/(\sigma^2 d)$.

Collecting all three terms at the $O(\log d/d)$ main order (setting aside $\mathcal{R}(z)$):

$$\|\nabla^2(\Phi - g)\|_2\big|_{\mathcal{G}} \le \frac{C_0^2\log d}{\sigma^2 d}\cdot\psi(u) + \sup_z \mathcal{R}(z),$$

where $\psi(u) = C_r\, e^{-u^2/2}(u^2 + 3u^4 + u^6)$. The coefficients track the three terms of Equation 25: u^2 from $g\nabla^2 Q$, $2u^4$ from the cross term, and $u^4 + u^6$ from $(Q-1)\nabla^2 g$ (where the u^6 part requires the pointwise $(1+u^2)$ Hessian bound). The envelope $\sup_{u\ge 0}\psi(u) = C_r \cdot M =: C''$ is finite, where $M := \sup_{u\ge 0} e^{-u^2/2}(u^2+3u^4+u^6) \approx 16.85$ (see Section C.5), giving $C'' \approx 27.3$. Therefore on $\mathcal{G}(z)$:

$$\|\nabla^2(\Phi - g)(z)\|_2 \le \frac{C_0^2 C''\log d}{\sigma^2 d} + \sup_z \mathcal{R}(z).$$

It remains to bound the bad-event contribution. We need a crude bound on $\|\nabla^2(\Phi - g)\|_2$ valid for *all* realizations of (D_2, D_3). Writing $x_k = \beta_k/\sigma$, $\Phi = \prod_k \cos x_k$, the second partial derivatives

satisfy $|\partial^2\Phi/\partial x_k^2| \le 1$ and $|\partial^2\Phi/\partial x_k\partial x_j| \le 1$ (each factor $|\cos|$ or $|\sin|$ is at most 1). Hence $\|\nabla_x^2\Phi\|_2 \le \|\nabla_x^2\Phi\|_F \le d$. Since $x = A^\top z/\sigma$ with orthogonal A, $\|\nabla_z^2\Phi\|_2 \le d/\sigma^2$. Combined with $\|\nabla^2 g\|_2 \le 1/\sigma^2$:

$$\|\nabla^2(\Phi - g)(z)\|_2 \le (d+1)/\sigma^2 =: B \quad \text{for all } (D_2, D_3).$$

Since (D_2, D_3) takes values in a finite set ($2^d \times 2^d$ Rademacher realizations), the expectation $h(z) = \mathbb{E}_{D_2,D_3}[\Phi(z) - g(z)]$ is a finite sum, and $\nabla^2 h(z) = \mathbb{E}_{D_2,D_3}[\nabla^2(\Phi - g)(z)]$ by termwise differentiation (no dominated convergence argument needed). The bad event contributes

$$\mathbb{E}\big[\|\nabla^2(\Phi - g)\|_2 \cdot \mathbf{1}_{\mathcal{G}^c}\big] \le B \cdot 2d^{-\alpha} = \frac{2(d+1)d^{-\alpha}}{\sigma^2} = O\left(\frac{d^{1-\alpha}}{\sigma^2}\right),$$

and for $\alpha > 2$, $d^{1-\alpha} = o(\log d/d)$, so this is absorbed into the good-event leading term.

Taking the supremum over $z \in \mathcal{X}_\Delta$ and collecting the three contributions — the good-event main term $C_0^2 C'' \log d/(\sigma^2 d)$, the $\|\nabla S\|^2$ remainder $\sup_z \mathcal{R}(z) = O(\log^2 d/d^2)$ (which for $d \ge d_4$ is at most $C_0^2 C'' \log d/(\sigma^2 d)$), and the bad-event $O(d^{1-\alpha}/\sigma^2)$ (which for $\alpha > 2$ and $d \ge d_3$ is also at most $C_0^2 C'' \log d/(\sigma^2 d)$) — gives

$$\sup_{z\in\mathcal{X}_\Delta} \|\nabla^2 h(z)\|_2 \le 3 \cdot \frac{C_0^2 C'' \log d}{\sigma^2 d} = \frac{C_H(\alpha)\log d}{\sigma^2 d},$$

where $C_H(\alpha) = 3C_0^2 C''$, and the factor of 3 accounts for the main term plus the two absorbed lower-order contributions, for $d \ge d_0 := \max(d_1, d_3, d_4)$. □

Lemma 7. *Define* $E(u,v) := \mathbb{E}[\tilde{k}(u,v)] - k(u,v) = h(u-v)$. *Then* $|\delta_{i,j}^2 E| \le \epsilon\, \delta\omega_{i,j}$ *with* $\epsilon = C_H \log d/(\sigma^2 d)$.

Proof. Since $E(u,v) = h(u-v)$ is shift-invariant, the same integral mean value theorem argument as in Lemma 4 applies with $\nabla^2 h$ replacing $\nabla^2 f$. The integration path $a + s\alpha - t\gamma$ lies in $\mathcal{X}_\Delta$ by convexity of $\mathcal{X}$, so Lemma 1 gives $\sup_z \|\nabla^2 h(z)\|_2 \le \epsilon$, yielding $|\delta_{i,j}^2 E| \le \epsilon\|\Delta x_i\|\|\Delta y_j\| = \epsilon\, \delta\omega_{i,j}$. □

C.3 Proof of Theorem 2

We prove by induction on m that

$$|B_{S,T}^{(m)}| := \left|\mathbb{E}\left[\tilde{K}_{S,T}^{(m)}\right] - K_{S,T}^{(m)}\right| \le \frac{[(L_k+\epsilon)^m - L_k^m]\,\omega_{S,T}^m}{(m!)^2}, \tag{26}$$

where $\epsilon = C_H \log d/(\sigma^2 d)$.

For $m = 1$, the first-order signature kernel is $K_{S,T}^{(1)} = \sum_{i=1}^S \sum_{j=1}^T \delta_{i,j}^2 k$ and its SORF approximation is $\tilde{K}_{S,T}^{(1)} = \sum_{i,j} \delta_{i,j}^2 \tilde{k}$. Since different (i,j) pairs use the same SORF matrix at level 1,

$$B_{S,T}^{(1)} = \sum_{i=1}^S \sum_{j=1}^T \delta_{i,j}^2 E,$$

where $E(u,v) = \mathbb{E}[\tilde{k}(u,v)] - k(u,v)$ is the base kernel bias. By Lemma 1 and the integral mean value theorem (as in the proof of Lemma 4, replacing $\nabla^2 k$ by $\nabla^2 E$), $|\delta_{i,j}^2 E| \le \epsilon\, \delta\omega_{i,j}$, so

$$|B_{S,T}^{(1)}| \le \epsilon \sum_{i,j} \delta\omega_{i,j} = \epsilon\, \omega_{S,T}.$$

Since $(L_k+\epsilon)^1 - L_k^1 = \epsilon$, the bound Equation 26 holds with $(m!)^2 = 1$.

Assume Equation 26 holds for $m-1$. The m-th order signature kernel satisfies

$$\tilde{K}_{S,T}^{(m)} = \sum_{i=1}^S \sum_{j=1}^T \tilde{K}_{i-1,j-1}^{(m-1)}\, \delta_{i,j}^2 \tilde{k}^{(m)},$$

where $\tilde{k}^{(m)}$ is the SORF approximation at level m, independent of levels $1, \dots, m-1$. Taking expectations and using the independence:

$$\mathbb{E}\Big[\tilde{K}_{S,T}^{(m)}\Big] = \sum_{i,j} \mathbb{E}\Big[\tilde{K}_{i-1,j-1}^{(m-1)}\Big]\, \mathbb{E}\Big[\delta_{i,j}^2 \tilde{k}^{(m)}\Big]\,.$$

Adding and subtracting to decompose the error:

$$\begin{aligned} B_{S,T}^{(m)} &= \sum_{i,j} \mathbb{E}[\tilde{K}_{i-1,j-1}^{(m-1)}]\, \mathbb{E}[\delta_{i,j}^2 \tilde{k}] - \sum_{i,j} K_{i-1,j-1}^{(m-1)}\, \delta_{i,j}^2 k \\ &= \underbrace{\sum_{i,j} B_{i-1,j-1}^{(m-1)}\, \delta_{i,j}^2 k}_{\text{(I)}} + \underbrace{\sum_{i,j} K_{i-1,j-1}^{(m-1)}\, \delta_{i,j}^2 E}_{\text{(II)}} + \underbrace{\sum_{i,j} B_{i-1,j-1}^{(m-1)}\, \delta_{i,j}^2 E}_{\text{(III)}}\,. \end{aligned} \tag{27}$$

We bound each term in absolute value. For (I), the inductive hypothesis combined with Lemma 4 gives

$$|\text{(I)}| \le \sum_{i,j} \frac{[(L_k+\epsilon)^{m-1} - L_k^{m-1}]\, \omega_{i-1,j-1}^{m-1}}{((m-1)!)^2} \cdot L_k\, \delta\omega_{i,j},$$

and Lemma 6 reduces this to

$$|\text{(I)}| \le L_k\, [(L_k+\epsilon)^{m-1} - L_k^{m-1}] \cdot \frac{\omega_{S,T}^m}{(m!)^2}.$$

For (II), the exact kernel satisfies $|K_{i-1,j-1}^{(m-1)}| \le L_k^{m-1}\, \omega_{i-1,j-1}^{m-1}/((m-1)!)^2$ (which follows by induction on the exact kernel using Lemma 4), and combined with $|\delta_{i,j}^2 E| \le \epsilon\, \delta\omega_{i,j}$,

$$|\text{(II)}| \le \epsilon\, L_k^{m-1} \cdot \frac{\omega_{S,T}^m}{(m!)^2}.$$

For (III), applying both the inductive hypothesis and the base kernel bias bound,

$$|\text{(III)}| \le \epsilon\, [(L_k+\epsilon)^{m-1} - L_k^{m-1}] \cdot \frac{\omega_{S,T}^m}{(m!)^2}.$$

Summing the three bounds,

$$\begin{aligned} |B_{S,T}^{(m)}| &\le \frac{\omega_{S,T}^m}{(m!)^2}\Big[L_k\big((L_k+\epsilon)^{m-1} - L_k^{m-1}\big) + \epsilon\, L_k^{m-1} + \epsilon\big((L_k+\epsilon)^{m-1} - L_k^{m-1}\big)\Big] \\ &= \frac{\omega_{S,T}^m}{(m!)^2}\Big[(L_k+\epsilon)\big((L_k+\epsilon)^{m-1} - L_k^{m-1}\big) + \epsilon\, L_k^{m-1}\Big] \\ &= \frac{\omega_{S,T}^m}{(m!)^2}\Big[(L_k+\epsilon)^m - (L_k+\epsilon)L_k^{m-1} + \epsilon\, L_k^{m-1}\Big] \\ &= \frac{[(L_k+\epsilon)^m - L_k^m]\, \omega_{S,T}^m}{(m!)^2}, \end{aligned} \tag{28}$$

where the last line uses $(L_k+\epsilon)L_k^{m-1} - \epsilon L_k^{m-1} = L_k^m$. This completes the induction. □

C.4 TRP Extension

We use the canonical RFSF-TRP construction of Tóth et al. [37, Def. 3.7, Alg. D.3], applied conditionally on RUBATO's SORF base random features. The argument below is a short adapter showing that this scaling is exactly RFSF-TRP applied per level, so that bias inherits Theorem 2 via the tower property; we do not reprove the underlying RFSF-TRP unbiasedness, which follows from Tóth et al. [37, Def. 3.7].

Construction. For each signature level $m \in \{1, \dots, M\}$, draw an independent matrix $P_m \in \mathbb{R}^{2D_{\mathrm{rf}} \times r}$ with i.i.d. $\mathcal{N}(0,1)$ entries ($r := d_{\mathrm{feat}}$), independent across levels and of the SORF randomness $\boldsymbol{D}$, matching the per-level Gaussian projection sampling step in Tóth et al. [37, Alg. D.3]. Given the canonical RFF-normalized SORF increments $\Delta\varphi_m^{(\ell)} \in \mathbb{R}^{2D_{\mathrm{rf}}}$ (cf. Section 3), set

$$\boldsymbol{\psi}_m^{\mathrm{TRP}}(x) \;=\; r^{-1/2} \sum_{i_1 < \cdots < i_m} \bigodot_{p=1}^{m} P_p^\top \Delta\varphi_p^{(i_p)}(x) \;\in\; \mathbb{R}^r, \tag{29}$$

and $\tilde{k}_{\mathrm{Sig}_m}^{\mathrm{TRP}}(x,y) = \langle \boldsymbol{\psi}_m^{\mathrm{TRP}}(x), \boldsymbol{\psi}_m^{\mathrm{TRP}}(y) \rangle$. This matches Tóth et al. [37, Def. 3.7] verbatim, with $\Delta\varphi_p^{(i_p)}$ produced by SORF rather than by dense Gaussian RFF.

Conditional unbiasedness and bias inheritance. Because $\mathbb{E}[(P_p^\top a)_k (P_p^\top b)_{k'}] = \langle a, b \rangle \mathbf{1}[k = k']$ for $a, b \in \mathbb{R}^{2D_{\mathrm{rf}}}$ and P_p are independent across levels, summing over $k = 1, \dots, r$ cancels the per-level r^{-1} factor in Equation 29. Specialising the RFSF-TRP kernel formula [37, Def. 3.7] to our SORF-built increments yields

$$\mathbb{E}_{P_{1:M}}\Big[\tilde{k}_{\mathrm{Sig}_m}^{\mathrm{TRP}}(x,y) \,\Big|\, \boldsymbol{D}\Big] \;=\; \sum_{\boldsymbol{i},\boldsymbol{j} \in \Delta_m} \prod_{p=1}^{m} \big\langle \Delta\varphi_p^{(i_p)}(x), \Delta\varphi_p^{(j_p)}(y) \big\rangle \;=\; \tilde{k}_{\mathrm{Sig}_m}^{\mathrm{SORF}}(x,y), \tag{30}$$

i.e. the RFSF-TRP estimator is conditionally unbiased for the RFSF kernel built from the same level-wise increments. Taking expectation over $\boldsymbol{D}$ and applying the tower property,

$$\mathbb{E}_{\boldsymbol{D},P}\Big[\tilde{k}_{\mathrm{Sig}_m}^{\mathrm{TRP}}\Big] \;=\; \mathbb{E}_{\boldsymbol{D}}\Big[\tilde{k}_{\mathrm{Sig}_m}^{\mathrm{SORF}}\Big],$$

so the per-level bias of the TRP estimator equals the per-level SORF bias and is therefore bounded by Theorem 2 with rate $O(\log d_H / d_H)$. (The corresponding RFSF/RFSF-TRP concentration result for *dense* Gaussian RFF increments is Tóth et al. [37, Thm. 3.2]; here we replace its base-RFF assumption by SORF and pay the $O(\log d_H / d_H)$ bias bound of Theorem 2.)

The level-0 constant $c_0 = 10^{-3}$ in Equation 5 is a numerical floor in the downstream attention feature, not part of this $m \geq 1$ kernel-bias statement.

C.5 Numerical Verification of Constants

The constant C'' in Lemma 1 involves the envelope function $\varphi(u) = e^{-u^2/2}(u^2 + 3u^4 + u^6)$, which arises from bounding the spectral norm of $\nabla^2 g(z)$ times polynomial terms in $u = \|z\|/\sigma$. The tighter coefficients compared to a naïve analysis stem from the key observation that $S(z) \leq 0$ (Lemma 5 (iv)), which gives $e^S \leq 1$ and $|e^S - 1| \leq |S|$ unconditionally, eliminating the need for the condition $|S| \leq 1/2$ and the associated factor of 2.

The envelope φ is bounded, unimodal, and decays super-exponentially for large u; its supremum $M := \sup_{u \geq 0} \varphi(u)$ is attained at a finite $u^\star$ determined by setting $\varphi'(u) = 0$ (a polynomial equation in u^2, solved numerically). The constants in Lemma 1 are then $C_r = (4/\pi)^2$ from the log-remainder bounds (Lemma 5); $C'' = C_r \cdot M$, the product of the log-remainder coefficient and the envelope supremum; $C_0 = \sqrt{2(\alpha+1)}$ for the chosen $\alpha > 2$; and $C_H(\alpha) = 3C_0^2 C''$, where the factor of 3 absorbs two lower-order remainders (the $\|\nabla S\|^2$ term and the bad-event contribution) into the leading term. All constants are finite and explicitly computable for any fixed $\alpha > 2$. The primary contribution of Theorem 2 is to establish the asymptotic rate $\epsilon = O(\log d / d)$, which guarantees consistency of the SORF signature kernel estimator as $d \to \infty$. The super-exponential decay of $\omega^m/(m!)^2$ in the truncation level m further ensures that the per-level bias remains summable. As shown in Section 5 (Figure 3b), the empirical pointwise bias decays as $\sim d^{-1.8}$ and falls within the noise floor of unbiased RFF for $d \geq 16$, consistent with the theoretical rate and well within the regime of practical RUBATO configurations. Tightening the constant C_H (e.g., via exact Rademacher moment-generating functions or exploiting the correlation structure of $\{\beta_k\}$ induced by the Hadamard matrix) is an interesting open direction noted in Section 6.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and Section 1 state the two challenges (representation and similarity), the proposed RUBATO architecture (Section 4), the SORF bias bound (Section 4.3), and the empirical comparison on six IMTS benchmarks (Section 5); each claim is supported by a corresponding section.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Section 6 contains a dedicated "Limitations" paragraph covering: (i) the SORF projection benefit emerges only at the larger end of the d_H range we test, so on benchmarks with low channel counts the wall-clock advantage is small; (ii) the proven $O(\log d_H / d_H)$ SORF signature kernel bias bound is loose against the empirical $\sim d_H^{-1.8}$ decay (Figure 3b), and a matching SORF variance bound is still missing; and (iii) the current decoder produces deterministic point forecasts and treats missingness only through mask channels rather than through an explicit mechanism. Section 4.3 also delineates the scope of the theoretical guarantee (base-kernel bias only; downstream TRP / cosine normalization / content-gated softmax not directly covered).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: Lemma 1, Theorem 2, and Corollary 3 are stated with their assumptions in Section 4.3; the formal restatement of the assumptions on compactness, kernel bandwidth, and Hadamard-padded dimension, and the complete proofs (with supporting lemmas and numerical constant verification) are given in Appendix C.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The full RUBATO architecture is specified in Section 4; data preprocessing, splits, and per-dataset normalization are described in Appendix B; hyperparameters and the training protocol (5 seeds, 20 epochs, patience 3, MSE loss) are listed in Section 5.1 and Appendix A.3; all baselines are cited with their original references.

   Guidelines:

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

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: An anonymized code archive containing the full RUBATO implementation, dataset preprocessing scripts, training entry points, and configuration files for all reported experiments is provided in the supplementary material, and an anonymized browsable mirror is available at `https://anonymous.4open.science/r/Rubato-A13C`; six public IMTS datasets used in our evaluation are referenced with their original sources in Appendix B.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 5.1 specifies the unified training protocol (MSE loss, 20 training epochs, early-stopping patience 3, 5 seeds); the six benchmarks and their irregularity/missingness characteristics are described in Appendix B, and the exact split, chunking, and preprocessing pipeline is provided in the anonymized code release; the per-dataset ablation grid (signature order M, bandwidth initialization σ_{init}, projection backbone) is in Appendix A.3.

Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: All forecasting (Table 1) and ablation (Table 2) results are reported as mean $\pm$ standard deviation across 5 independent random seeds, with seeds drawn deterministically from a fixed base seed; the standard deviation is computed across the 5 per-seed runs and reported in the same units as the mean.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: All experiments are run on a single NVIDIA RTX 3090 GPU; per-epoch wall-clock measurements (SORF vs. dense RFF) are tabulated in Table 2 for USHCN, FNSPID, and MIMIC; the projection-stage GPU benchmarks across d_H are reported in Figure 3 of the main text.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research uses publicly available, de-identified IMTS benchmark datasets with their original licenses and access protocols, does not involve human subjects research or new data collection, and adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: The work is foundational research on irregular multivariate time series forecasting, with applications such as clinical monitoring and climate forecasting that depend on the deployment context rather than on the architecture itself; we do not introduce a deployable model with direct societal impact, so a separate broader-impact analysis is not warranted at this stage.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We do not release any pre-trained generative models, scraped datasets, or other assets with a foreseeable high risk of misuse; all evaluation uses pre-existing public IMTS benchmarks under their established access protocols.

Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Each dataset (MIMIC-III, PhysioNet, USHCN, FNSPID, EPA-AIR, ClusterTrace) and each baseline (SeFT, PrimeNet, mTAN, NeuralFlows, t-PatchGNN, KAFNet, APN, Hi_Patch, GraFITi, HyperIMTS) is cited at its original publication; the established access conditions for each benchmark are followed (e.g., credentialed access for MIMIC-III), and we do not redistribute any of these assets.

    Guidelines:

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

13. **New assets**

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

    Answer: [Yes]

    Justification: The new asset accompanying the paper is the anonymized RUBATO code release in the supplementary material; it includes a README with the exact command, environment, and configuration files needed to reproduce all reported results, plus per-dataset preprocessing scripts.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects; all evaluation uses pre-existing public IMTS benchmark datasets.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [N/A]

    Justification: The paper does not conduct research with human subjects; the clinical IMTS benchmarks (MIMIC-III, PhysioNet) are reused under their established access and ethical review procedures, with which we comply.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs are not used as a component of the core methodology, theoretical analysis, or experimental pipeline in this research; any incidental writing or formatting assistance does not affect the scientific rigor or originality of the work.

    Guidelines:

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