PARSI: PARALLEL TEMPORAL MODELING WITH SELECTIVE INTERACTION FOR IRREGULAR MULTIVARIATE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Irregular multivariate time series (IMTS) arise when multiple variables are measured at different times and with varying gaps between observations, as commonly occurs in healthcare and finance. These irregularities make it difficult to model how individual variables evolve and how they relate to one another, giving rise to two key challenges: (1) preserving useful short- and long-range temporal patterns that a single temporal resolution or progressive aggregation may obscure, and (2) capturing changing cross-variable dependencies without computing full attention for every variable representation. To address these challenges, we propose PARSI, a forecasting framework that combines parallel temporal modeling with selective cross-variable interaction. First, PARSI encodes different lengths of each variable's history in parallel and adaptively combines the resulting representations, preserving complementary temporal information. Second, it estimates whether cross-variable dependencies are concentrated on particular variables or distributed broadly, and uses this estimate to select representations for full cross-variable attention. The remaining representations receive a shared global summary, reducing computation while retaining access to information from other variables. Extensive experiments on real-world IMTS forecasting datasets demonstrate that PARSI achieves up to 10.1% relative improvement in forecasting accuracy over representative state-of-the-art baselines while reducing inference time by 25.0%.

1 INTRODUCTION

Irregular multivariate time series (IMTS) forecasting is an important task in real-world applications, including financial analysis and environmental monitoring (Dong et al., 2024; Bhandari et al., 2018). In these applications, multiple variables are measured to characterize the evolution of a system, but their observations rarely follow a common, regular schedule. Different sensing mechanisms, event-driven measurements, and missing observations produce two defining characteristics: irregular intervals between successive observations of the same variable and asynchronous observation times across different variables. These characteristics distinguish IMTS from the regularly sampled sequences assumed by most multivariate forecasting models (Lim & Zohren, 2021). Mapping observations to a common time grid makes such models applicable, but may expand the input sequence, obscure the original observation times, and introduce artifacts through alignment or imputation (Shukla & Marlin, 2019). These limitations motivate forecasting methods that explicitly account for irregular observations.

Beyond accommodating irregular sampling, accurate forecasting requires learning two types of dependencies: temporal dependencies within each variable and relationships across variables. A future value may depend on both recent changes and patterns extending over a longer history, while observations of other variables may provide additional predictive information. These requirements are closely related: the historical information retained for each variable determines what information is available for subsequent cross-variable interaction. However, retaining useful temporal patterns and exchanging information across variables both become more demanding as the observation history and the number of variables increase. This gives rise to two central challenges in IMTS forecasting.

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

The first challenge is how to capture patterns from different lengths of observation history without obscuring their distinct temporal information. Recent observations can reveal local changes, whereas a longer history can provide the context needed to interpret those changes and anticipate future behavior (Wu et al., 2023; Li et al., 2026). For irregular data, these sources of information must also be interpreted in relation to their observation times: a fixed number of observations does not necessarily cover a fixed duration. An effective temporal representation should therefore accommodate irregular timing while retaining useful patterns from both shorter and longer historical ranges. Recent approaches have made progress toward this goal through temporal patching and aggregation. For example, Zhang et al. (2024a) developed a transformable patching graph neural network that maintains a unified temporal resolution across variables, avoiding sequence-length explosion when the number of variables is large. Liu et al. (2026b) introduced time-aware patch aggregation that adapts temporal patches to irregular observation patterns, providing greater flexibility in capturing dependencies at different scales. Nevertheless, organizing history at a single resolution can limit the temporal patterns available to the model, while progressive aggregation can merge fine-grained information before its relevance to forecasting is determined. The issue is not simply whether a model covers a sufficiently long history, but whether distinct views of that history remain available when forming the final representation. This motivates modeling multiple temporal ranges in parallel and learning how to combine them rather than relying exclusively on a single resolution or a sequence of increasingly aggregated representations.

The second challenge is how to capture changing cross-variable dependencies while avoiding unnecessary interaction costs. The relevance of one variable to another can change with the observed conditions. Consequently, different variable representations need not exhibit the same pattern of cross-variable dependence. Some may assign high relevance to a small subset of variables and therefore benefit from an individually weighted aggregation of information. Others may have more broadly distributed dependencies, for which a shared summary of the system can serve as a less expensive alternative. Here, a variable state refers to the representation of a variable under the current observation history, rather than a fixed property of the variable itself. Recent graph- and hypergraph-based methods have explored dynamic cross-variable interactions for IMTS forecasting (Li et al., 2025; Liu et al., 2026a). These developments highlight the importance of allowing relationships to depend on the observations rather than treating them as fixed. However, adapting interaction weights does not by itself determine where detailed interaction is most useful. Applying full global attention to every variable state computes a separate aggregation over all variables, regardless of whether that state has a distinctive dependency pattern or can be adequately represented using shared information. This motivates a selective mechanism that identifies states likely to benefit from individualized global aggregation and gives the remaining states access to a shared global context. Importantly, selection concerns which states receive a separately computed aggregation, not which variables are permitted to contribute information.

To address these challenges, we propose PARSI, a forecasting framework combining parallel temporal modeling with selective cross-variable interaction. PARSI encodes observations using their values, variable identities, and temporal information. For the first challenge, independent branches model different historical ranges of each variable, then feature-wise adaptive fusion combines their outputs into a variable-level representation. For the second, PARSI uses attention concentration to selectively assign individualized global attention to variable states. Finally, a decoder combines the historical representation with target timestamps to generate forecasts.

This paper advances the state of the art in the following ways. First, we propose PARSI, an IMTS forecasting framework that integrates multi-range temporal modeling with selective cross-variable interaction, capturing diverse historical patterns while limiting interaction costs. Second, we design parallel temporal branches to capture patterns across different historical ranges and a selective attention mechanism to reduce the cost of modeling cross-variable dependencies. Finally, extensive experiments on real-world IMTS forecasting datasets demonstrate that PARSI achieves up to 10.1% relative improvement in forecasting accuracy over representative state-of-the-art baselines.

2 Related Work

Deep learning approaches to IMTS forecasting seek to account for irregular observation times while learning dependencies within and across variables. We review related work from two complemen-

tary perspectives: temporal modeling, which concerns how irregular histories are represented to capture temporal patterns, and cross-variable dependency modeling, which concerns how information is exchanged across variables or observations.

Temporal Modeling for Irregular Time Series: Representing observations recorded at unequal time intervals is a central problem in IMTS forecasting. Continuous-time methods address this problem by modeling how a system evolves between observations. For example, Latent ODE (Rubanova et al., 2019) represents latent dynamics through continuous-time evolution, while GRU-ODE Bayes (De Brouwer et al., 2019) combines continuous evolution with discrete updates when observations arrive. NeuralFlows (Biloš et al., 2021) and CRU (Schirmer et al., 2022) provide alternative formulations for modeling irregular trajectories without first aligning observations to a shared time grid. These methods accommodate irregular observation times through continuous-time evolution, but do not explicitly separate historical information into range-specific representations. PARSI instead models different temporal ranges through independent parallel branches and combines them using feature-wise adaptive fusion. Another line of work organizes irregular observations into temporal patches or aggregated representations. For example, t-PatchGNN (Zhang et al., 2024a) uses a predefined patch resolution, whereas APN (Liu et al., 2026b) adapts aggregation boundaries to local observation patterns. Hi-Patch (Luo et al., 2025) and ReIMTS (Li et al., 2026) further incorporate information at multiple temporal resolutions through hierarchical or recursive construction. These methods illustrate different ways to organize irregular histories, where representations at different resolutions are constructed progressively, so later representations depend on earlier transformations or aggregations. PARSI instead constructs representations for different temporal ranges in parallel, avoiding progressive dependence across resolutions.

Cross-Variable Dependency Modeling: In addition to each variable's own history, observations of other variables can provide useful information for forecasting. Existing methods capture these relationships through several interaction mechanisms. For example, GNeuralFlow (Mercatali et al., 2024) couples continuous dynamics with learned variable interactions. HyperIMTS (Li et al., 2025) uses temporal and variable hyperedges to capture higher-order relationships, while ASTGI (Liu et al., 2026a) constructs adaptive interactions among individual observations. KAFNet (Zhou et al., 2026) models global correlations through frequency-domain linear attention. These approaches explore how to define relationships and exchange information across variables or observations. However, computing a separate global aggregation for every state can introduce unnecessary overhead. PARSI therefore allocates query-specific global attention selectively while retaining shared global information for the remaining states.

3 METHODOLOGY

In this section, we present PARSI for IMTS forecasting. We first formulate the forecasting task and introduce the framework and observation encoding. We then describe its two core components: parallel multi-range temporal modeling and selective cross-variable interaction. Finally, we introduce the query-conditioned decoder for prediction at the requested future timestamps.

3.1 PROBLEM FORMULATION

Given historical observations collected at irregular and potentially asynchronous times, we describe the forecasting input through variable specific histories and specify the desired outputs through future query timestamps. In more detail, for a series with N variables, the available history of variable n is $\mathcal{X}^{(n)} = \{(t_i^{(n)}, x_i^{(n)})\}_{i=1}^{L_n}$, where L_n is its number of observations, and $t_i^{(n)} \in \mathbb{R}$ and $x_i^{(n)} \in \mathbb{R}$ are the timestamp and value of its i-th observation, respectively, ordered chronologically. Together, these variable-specific histories form the complete input $\mathcal{X} = \{\mathcal{X}^{(n)}\}_{n=1}^{N}$. Keeping a separate timestamp sequence for each variable allows $\mathcal{X}$ to capture both irregular sampling within variables and asynchronous observations across variables.

Given this history, we seek predictions at specified future timestamps rather than on a fixed, shared prediction grid. For variable n, these timestamps are denoted by $\mathcal{Q}^{(n)} = \{q_j^{(n)}\}_{j=1}^{Q_n}$, where Q_n is the number of requested predictions and the queries are ordered with $q_1^{(n)}$ being the earliest. The complete query collection is $\mathcal{Q} = \{\mathcal{Q}^{(n)}\}_{n=1}^{N}$. Each query $q_j^{(n)}$ is associated with a predicted value

$\hat{x}_j^{(n)}$. Hence, we let $\mathcal{X}$ specify the available evidence and $\mathcal{Q}$ denote which future values the model should predict.

To provide these variable-length histories as tensor inputs to the model, we represent $\mathcal{X}$ using padded matrices $(\mathbf{X}, \mathbf{T}, \mathbf{M})$, in which $\mathbf{X}, \mathbf{T} \in \mathbb{R}^{L \times N}$ store observation values and timestamps, $\mathbf{M} \in \{0,1\}^{L \times N}$ distinguishes valid observations from padding, and $L = \max_n L_n$. Each column corresponds to one variable: for $i \leq L_n$, we set $X_{i,n} = x_i^{(n)}$, $T_{i,n} = t_i^{(n)}$, and $M_{i,n} = 1$, and the remaining positions have $M_{i,n} = 0$. Padding equalizes the column lengths without aligning their timestamps. Consequently, row i denotes the i-th observation of each variable, not necessarily observations made at the same time.

Using this computational representation of $\mathcal{X}$, the forecasting task is to learn a parameterized mapping from the observed histories and future queries to the corresponding predictions, i.e.,

$$\widehat{\mathcal{Y}} = F_\theta(\mathbf{X}, \mathbf{T}, \mathbf{M}, \mathcal{Q}), \qquad \widehat{\mathcal{Y}} = \left\{ [\hat{x}_j^{(n)}]_{j=1}^{Q_n} \right\}_{n=1}^{N}, \tag{1}$$

where θ is the model parameters, and $\hat{x}_j^{(n)}$ is the forecast for variable n at query timestamp $q_j^{(n)}$.

3.2 Framework Overview and Observation Encoding

The forecasting task in Equation 1 requires a model to connect variable-specific, irregular observation histories to predictions at specified future timestamps. As illustrated in Figure 1, PARSI addresses this problem through a sequence of complementary stages. First, an observation encoder embeds each measurement together with its timing and variable identity. Parallel temporal branches then process these embeddings over different within-window interaction ranges, and adaptive fusion combines their outputs into one historical representation per variable. These variable-level representations provide the input to selective cross-variable interaction, which supplements each variable's own history with information from other variables. Finally, a query-conditioned decoder converts the resulting representations into predictions at the timestamps specified by $\mathcal{Q}$. The representation flow is therefore $(\mathbf{X}, \mathbf{T}, \mathbf{M}) \longrightarrow \mathbf{H}^O \longrightarrow \mathbf{H}^T \longrightarrow \mathbf{H}^I \longrightarrow \widehat{\mathcal{Y}}$, with the query timestamps providing the temporal reference for encoding and the requested prediction times for decoding.

Irregular observation encoding: We begin by constructing $\mathbf{H}^O$ from the padded histories. As established in the problem formulation, the row index of these histories indicates observation order rather than a shared timestamp. Consequently, an observation's position alone does not specify either its distance from the forecast period or its temporal separation from neighboring measurements. We explicitly encode both quantities so that subsequent temporal modeling has access to the original irregular timing. For the i-th valid observation of variable n, we use $q_1^{(n)} - T_{i,n}$ to describe its distance to the earliest forecasting query. We additionally define $\delta_i^{(n)} = T_{i,n} - T_{i^-,n}$ to describe the elapsed time since the preceding valid observation, where i^- indexes that observation. For the first valid observation, we set $\delta_i^{(n)} = 0$.

The observation value and these temporal quantities are mapped into a shared d-dimensional embedding space $\mathbf{e}_i^{x,(n)} = \phi_x([X_{i,n}, M_{i,n}])$ and $\mathbf{e}_i^{t,(n)} = \phi_t\Big([q_1^{(n)} - T_{i,n}, \delta_i^{(n)}]\Big)$, where ϕ_x and ϕ_t are learnable projections shared across variables. To distinguish measurements belonging to different variables, we also introduce a learnable variable embedding $\mathbf{e}^{v,(n)}$. The three embeddings are combined as

$$\mathbf{h}_i^{(n)} = M_{i,n}\ \mathrm{LN}\Big(\mathbf{e}_i^{x,(n)} + \mathbf{e}_i^{t,(n)} + \mathbf{e}^{v,(n)}\Big), \tag{2}$$

where LN denotes layer normalization. By multiplying the embeddings by $M_{i,n}$, we set padded positions to zero and use only valid observations to construct the temporal windows below. Collecting these embeddings yields $\mathbf{H}^O \in \mathbb{R}^{L \times N \times d}$. This representation retains a separate observation sequence for each variable, now augmented with value, timing, and identity information. The next stage aggregates these observation-level embeddings into variable-level historical representations.

3.3 Parallel Multi-Range Temporal Modeling

The encoded observations in $\mathbf{H}^O$ preserve irregular timing, but forecasting also requires a summary of how observations relate over time. Choosing a single temporal interaction range introduces

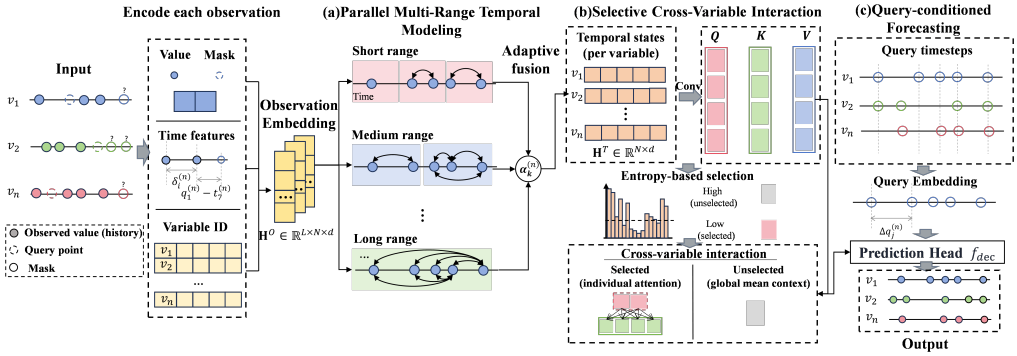

Figure 1: Overview of PARSI.

a trade-off under uneven observation density: short windows retain a local view but may contain few measurements, whereas longer windows allow broader interaction before observations are compressed. PARSI therefore constructs K complementary temporal views directly from $\mathbf{H}^O$. Each branch aggregates observations within its own temporal windows and remains independent until fusion. Thus, a longer-range view is computed from the original observation embeddings rather than from summaries already compressed by a shorter-range branch.

Temporal range construction: To define interaction ranges in elapsed time rather than observation count, we partition the historical interval using timestamp-based windows. Let t_s be the earliest valid timestamp across the historical observations. For variable n, branch k divides $[t_s, q_1^{(n)})$ into W_k equal-duration windows:

$$\mathcal{W}_{k,p}^{(n)} = \left[t_s + \frac{p-1}{W_k}(q_1^{(n)} - t_s),\ t_s + \frac{p}{W_k}(q_1^{(n)} - t_s) \right), \quad p = 1, \ldots, W_k. \tag{3}$$

For a given variable, all branches cover the same interval but differ in how they partition it. We gather the valid observation embeddings whose timestamps fall in $\mathcal{W}_{k,p}^{(n)}$ and arrange them chronologically $\mathbf{U}_{k,p}^{(n)} = [\mathbf{h}_{i_1}^{(n)}, \ldots, \mathbf{h}_{i_{L_{k,p}^{(n)}}}^{(n)}] \in \mathbb{R}^{L_{k,p}^{(n)} \times d}$, where $L_{k,p}^{(n)}$ is the number of observations in that window.

Within-window temporal attention: Having defined which observations may interact, we next summarize their temporal relationships within each nonempty window. Causal 1-D convolutions Bai et al. (2018); Wang et al. (2025) construct queries and keys from local observation sequences, while a linear projection constructs values:

$$\mathbf{Q}_{k,p}^{(n)} = \mathrm{Conv}_Q\left(\mathbf{U}_{k,p}^{(n)}\right), \quad \mathbf{K}_{k,p}^{(n)} = \mathrm{Conv}_K\left(\mathbf{U}_{k,p}^{(n)}\right), \quad \mathbf{V}_{k,p}^{(n)} = \mathbf{U}_{k,p}^{(n)}\mathbf{W}_V. \tag{4}$$

The convolutional projections capture local observation patterns, while the encoded inputs preserve irregular temporal information. Within-window attention then produces

$$\mathbf{A}_{k,p}^{(n)} = \mathrm{Softmax}\left(\frac{\mathbf{Q}_{k,p}^{(n)} \mathbf{K}_{k,p}^{(n)\top}}{\sqrt{d_h}} + \mathbf{M}_{\mathrm{causal}} \right) \mathbf{V}_{k,p}^{(n)}, \tag{5}$$

where d_h is the attention-head dimension and $\mathbf{M}_{\mathrm{causal}}$ prevents each observation from attending to subsequent observations in the window. After the residual update $\widetilde{\mathbf{U}}_{k,p}^{(n)} = \mathbf{A}_{k,p}^{(n)} + \mathbf{U}_{k,p}^{(n)}$, we retain the last valid state:

$$\mathbf{z}_{k,p}^{(n)} = \begin{cases} \widetilde{\mathbf{U}}_{k,p}^{(n)}[L_{k,p}^{(n)}], & L_{k,p}^{(n)} > 0, \\ \mathbf{0}, & L_{k,p}^{(n)} = 0. \end{cases} \tag{6}$$

This state is used as the window summary because, under causal attention, it can incorporate all preceding observations in that window. An empty window contributes a zero vector. Note that the window summaries capture separate portions of the history. To obtain a representation of the entire interval, we concatenate them in temporal order and apply a branch-specific projection $\mathbf{h}_k^{T,(n)} =$

$\phi_k(\mathbf{z}_{k,1}^{(n)} \| \cdots \| \mathbf{z}_{k,W_k}^{(n)}) \in \mathbb{R}^d$ where $\|$ denotes concatenation. Each $\mathbf{h}_k^{T,(n)}$ thus summarizes the same variable history, but with a different range of within-window interaction before compression.

Feature-wise adaptive fusion: The parallel branches provide complementary temporal representations, but their relevance need not be identical for every variable or feature. We therefore learn their contributions from the observed history instead of averaging them with fixed weights. For each branch, a learnable mapping produces a score vector $\mathbf{s}_k^{(n)} = \psi_k(\mathbf{h}_k^{T,(n)}) \in \mathbb{R}^d$. For feature dimension ℓ, we normalize these scores across branches $\alpha_{k,\ell}^{(n)} = \frac{\exp(s_{k,\ell}^{(n)})}{\sum_{r=1}^{K} \exp(s_{r,\ell}^{(n)})}$. Unlike a single weight per branch, this feature-wise weighting allows different dimensions of a variable's representation to emphasize different temporal ranges. The weighted combination and its residual refinement are

$$\bar{\mathbf{h}}_n^T = \sum_{k=1}^{K} \boldsymbol{\alpha}_k^{(n)} \odot \mathbf{h}_k^{T,(n)}, \quad \mathbf{h}_n^T = \mathrm{LN}\big(\bar{\mathbf{h}}_n^T + \phi_f(\bar{\mathbf{h}}_n^T)\big)\,. \tag{7}$$

where $\odot$ denotes element-wise multiplication and ϕ_f is a feed-forward transformation. Stacking the resulting states gives $\mathbf{H}^T \in \mathbb{R}^{N \times d}$. This yields one fixed-dimensional state per variable. However, each state still depends only on its own variable's observations. We then model cross-variable relationships at this level without observation-level temporal alignment.

3.4 Selective Cross-Variable Interaction

The temporal states in $\mathbf{H}^T$ provide a common representation space in which variables can exchange information despite having different observation counts and timestamps. Full attention could enrich every state through aggregation over all variables, but computing all pairwise affinities may be unnecessary when some states receive similar global context. Related query selection has been explored in ProbSparse attention (Zhou et al., 2021). PARSI therefore distinguishes between shared global context and individualized cross-variable aggregation. We use sampled attention entropy as a heuristic to identify states that require individualized aggregation, while the remaining states receive the shared mean context.

Entropy-based state selection: We first map $\mathbf{H}^T$ into attention queries, keys, and values using independent 1×1 convolutions $\mathbf{Q} = \phi_Q(\mathbf{H}^T)$, $\mathbf{K} = \phi_K(\mathbf{H}^T)$ and $\mathbf{V} = \phi_V(\mathbf{H}^T)$. Here, each attention query corresponds to a variable state seeking information from other states. Computing attention entropy over all keys would itself require the full pairwise affinity matrix. To avoid this, we randomly sample S keys, $\mathbf{K}' = \{\mathbf{k}'_j\}_{j=1}^{S}$, and use them to estimate the concentration of each query's affinities. For query $\mathbf{q}_i$, we compute

$$p_{ij} = \frac{\exp(\mathbf{q}_i^\top \mathbf{k}'_j)}{\sum_{r=1}^{S} \exp(\mathbf{q}_i^\top \mathbf{k}'_r)}, \qquad E_i = -\sum_{j=1}^{S} p_{ij} \log p_{ij}, \tag{8}$$

where E_i indicates that a query concentrates its sampled attention on fewer keys. Let the index set $\mathcal{I}$ contains the K_s lowest-entropy queries with $K_s = \min(N,\ c\,\lceil \log N \rceil)$, and c controls the selection budget.

Selective global aggregation: After selection, each state in $\mathcal{I}$ attends to all variables using its own attention weights. The remaining states receive the mean value representation:

$$\mathbf{c}_i = \begin{cases} \displaystyle\sum_{j=1}^{N} a_{ij} \mathbf{v}_j, & i \in \mathcal{I}, \\ \displaystyle\frac{1}{N} \sum_{j=1}^{N} \mathbf{v}_j, & i \notin \mathcal{I}, \end{cases} \qquad a_{ij} = \mathrm{Softmax}_j\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d_h}}\right) \quad (i \in \mathcal{I}). \tag{9}$$

Importantly, the sampled keys are used only for selection. The selected states aggregate over the complete set of variables, and all variables also contribute to the shared mean context. Selection therefore restricts the recipients of individualized attention, not the available information sources.

To incorporate cross-variable information without replacing each variable's own historical representation, we add the context through a residual path. Stacking $\mathbf{c}_i$ into $\mathbf{C}$ gives

$$\mathbf{H}^I = \rho(\mathbf{H}^T + \mathbf{C}) = [\mathbf{h}_1^I, \ldots, \mathbf{h}_N^I]^\top \in \mathbb{R}^{N \times d}, \tag{10}$$

Table 1: Forecasting performance on representative IMTS datasets. Results are reported as mean ± standard deviation over repeated runs. Lower values are better. Best results are shown in **bold**, and second-best results are underlined.

Method	FNSPID		CESNET		ClusterTrace		USHCN	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
GRU-D	0.1156±0.0054	0.0358±0.0022	0.6093±0.0010	0.6827±0.0009	1.1056±0.0025	2.7170±0.0187	0.2872±0.0070	0.2242±0.0405
CRU	0.1675±0.0132	0.0652±0.0083	0.6154±0.0023	0.6918±0.0028	1.0242±0.0620	2.2009±0.2364	0.3248±0.0120	0.2438±0.0109
NeuralFlow	0.1388±0.0082	0.0499±0.0039	0.6139±0.0015	0.6887±0.0010	1.0701±0.0131	2.6300±0.0581	0.3353±0.0075	0.4168±0.0429
SeFT	0.5692±0.0061	0.7292±0.0156	0.7245±0.0056	0.8513±0.0081	1.0464±0.0014	2.4442±0.0053	0.4441±0.0019	0.4098±0.0003
mTAN	0.1319±0.0051	0.0444±0.0017	0.6150±0.0005	0.6905±0.0011	0.9953±0.0290	2.3481±0.1958	0.5503±0.0293	0.6589±0.0825
DLinear	0.1060±0.0006	0.0343±0.0003	0.6275±0.0001	0.7161±0.0001	0.9972±0.0108	2.4442±0.0892	0.3775±0.0052	0.3977±0.0227
PatchTST	0.1041±0.0018	0.0332±0.0003	0.6140±0.0012	0.6993±0.0013	0.8912±0.0177	1.7172±0.1578	0.3419±0.0329	0.2509±0.0386
TimeMixer	0.1364±0.0083	0.0468±0.0054	0.6167±0.0012	0.6970±0.0011	0.8330±0.0093	1.4628±0.0571	0.4201±0.1625	0.5132±0.4902
t-PatchGNN	0.1214±0.0089	0.0388±0.0038	0.6136±0.0018	0.6914±0.0025	0.8858±0.0337	1.8745±0.1366	0.2877±0.0245	0.1772±0.0243
APN	0.0976±0.0047	0.0303±0.0016	0.6208±0.0030	0.7056±0.0028	0.9070±0.0061	1.9270±0.0359	0.2612±0.0150	0.1591±0.0123
Warpformer	0.0976±0.0033	0.0303±0.0011	0.6084±0.0009	0.6829±0.0010	0.9015±0.0208	1.9182±0.0512	0.2724±0.0045	0.1844±0.0055
ASTGI	0.1009±0.0008	0.0313±0.0004	0.6036±0.0016	0.6760±0.0010	0.7687±0.0155	1.4266±0.0889	0.2611±0.0141	0.1652±0.0115
HyperIMTS	0.1139±0.0051	0.0367±0.0030	0.6146±0.0017	0.6911±0.0006	0.8993±0.0181	1.7419±0.0786	0.2811±0.0265	0.1775±0.0121
Hi-Patch	0.0945±0.0022	0.0290±0.0007	**0.6028±0.0015**	0.6743±0.0017	0.8063±0.0386	**1.3519±0.1362**	0.2745±0.0224	0.1676±0.0148
PARSI (Ours)	**0.0922±0.0027**	**0.0281±0.0005**	0.6033±0.0022	**0.6742±0.0017**	**0.7557±0.0055**	1.3655±0.0270	**0.2482±0.0139**	**0.1506±0.0093**

where ρ is a GLU-based residual refinement block. The resulting $\mathbf{H}^I$ combines within-variable temporal information with cross-variable context. It provides the historical representation needed for forecasting, but does not yet specify the requested prediction time. The final stage introduces that information through query-conditioned decoding.

3.5 Query-Conditioned Forecasting

As defined in Section 3.1, outputs are indexed by future timestamps for each variable, so the decoder must combine each historical state $\mathbf{h}_n^I$ with the timestamp of the value being predicted. Then, we express each query relative to the earliest forecasting timestamp of its variable $\Delta q_j^{(n)} = q_j^{(n)} - q_1^{(n)}$. This uses the same temporal anchor as the observation encoder. A learnable temporal mapping embeds the query offset $\mathbf{e}_j^{q,(n)} = \phi_q(\Delta q_j^{(n)})$, and a regression head combines it with the corresponding variable state $\hat{x}_j^{(n)} = f_{\text{dec}}(\mathbf{h}_n^I \| \mathbf{e}_j^{q,(n)})$, where ϕ_q is the temporal mapping and f_{dec} is the regression head. For a given variable, the historical state is shared across its requested predictions, while the query embedding specifies the time associated with each output. Applying this decoder to all timestamps in $\mathcal{Q}$ produces $\widehat{\mathcal{Y}}$, completing the mapping F_θ defined in Equation 1.

4 Experiments

We evaluate PARSI to answer three questions: (1) Does it provide consistently competitive forecasting performance across different IMTS datasets? (2) How do parallel multi-range temporal modeling, adaptive fusion, and cross-variable interaction contribute to its performance? (3) How sensitive is it to the number of temporal branches and additional observation loss? We first compare PARSI with representative forecasting baselines, then isolate the contributions of its main components through ablation studies. Finally, we examine branch-count sensitivity, robustness to observation sparsity and computational efficiency. Due to the page limit, additional experimental details and supplementary analyses are provided in the supplementary material.

4.1 Experimental Setup

Dataset. We evaluate PARSI on four real-world datasets from TIME-IMM (FNSPID, CESNET and ClusterTrace) and one traditional IMTS benchmark (i.e. USHCN). This collection allows us to examine whether the proposed model generalizes beyond a single dataset or observation pattern. To ensure fairness and comparability of the results, all datasets are preprocessed by strictly following the standard procedures established in prior state-of-the-art works (Yalavarthi et al., 2024; Zhang et al., 2024a; Chang et al., 2025; Zhang et al., 2024b). We uniformly split the TIME-IMM datasets into training, validation, and test sets with a ratio of 60%, 20%, and 20%, respectively, while retaining the predefined splits for USHCN dataset.

Table 2: Ablation study on temporal modeling, inter-variable interaction, and adaptive fusion.

Ablation	Metric	FNSPID	CESNET	ClusterTrace	USHCN
w/o Multi-Range Temporal	MAE	0.1086±0.0037	0.6131±0.0018	0.8828±0.0059	0.2753±0.0233
	MSE	0.0337±0.0013	0.6880±0.0006	1.7621±0.0379	0.1593±0.0124
w/o Inter-Variable Interaction	MAE	0.0924±0.0025	0.6037±0.0017	0.7560±0.0081	0.2632±0.0081
	MSE	0.0286±0.0005	0.6748±0.0013	1.3674±0.0385	0.1553±0.0155
w/o Adaptive Fusion	MAE	0.0925±0.0018	0.6061±0.0016	0.7648±0.0031	0.2811±0.0201
	MSE	0.0282±0.0002	0.6786±0.0018	1.4176±0.0234	0.1717±0.0384
PARSI (Ours)	MAE	**0.0922±0.0027**	**0.6033±0.0022**	**0.7557±0.0055**	**0.2482±0.0139**
	MSE	**0.0281±0.0005**	**0.6742±0.0017**	**1.3655±0.0270**	**0.1506±0.0093**

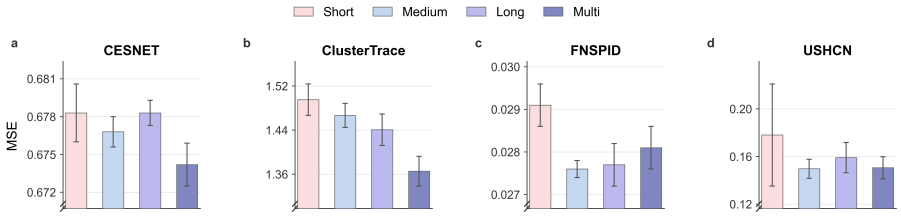

Figure 2: Performance comparison under different temporal ranges.

Baseline. We compare PARSI with 14 baselines covering several modeling paradigms. These include GRU-D (Che et al., 2018), CRU (Schirmer et al., 2022), and NeuralFlow (Biloš et al., 2021), as well as the set-based method SeFT (Horn et al., 2020) and the attention-based method mTAN (Shukla & Marlin, 2021). We also include APN (Liu et al., 2026b), t-PatchGNN (Zhang et al., 2024a), ASTGI (Liu et al., 2026a), and Warpformer (Zhang et al., 2023) to cover additional approaches to IMTS representation learning. Hi-Patch (Luo et al., 2025) and HyperIMTS (Li et al., 2025) provide comparisons with recent multi-scale temporal modeling methods. Finally, we evaluate three models originally developed for regularly sampled time series: DLinear (Zeng et al., 2023), PatchTST (Nie et al., 2023), and TimeMixer (Wang et al., 2024).

Implementation and evaluation. We implement all experiments in PyTorch 2.6 with CUDA 12.4 and run them on a server equipped with an NVIDIA GeForce 4090 GPU. For the baselines, we use the hyperparameter settings reported in the original papers as starting points and further tune selected hyperparameters for a subset of models on the validation set. We evaluate forecasting performance using MAE and MSE and report the mean and standard deviation over five independent runs.

4.2 Main Results

Table 1 compares PARSI with 14 baselines on four representative datasets. PARSI achieves the best MAE and MSE on FNSPID and USHCN. Compared to the strongest baseline for each metric, PARSI achieves strong reductions of approximately MAE by 4.9% and MSE by 5.3%, respectively. Although PARSI ranks second in the remaining two evaluations, it offers a favorable accuracy–efficiency trade-off. On CESNET, dense observations and scheduling jitter may limit the benefits of multi-range modeling. Nevertheless, PARSI nearly matches Hi-Patch's MAE (0.6033 vs. 0.6028) and achieves the best MSE. On ClusterTrace, resource-dependent sampling may hinder temporal modeling, potentially explaining its lower MAE but slightly higher MSE than Hi-Patch. Overall, PARSI remains competitive while reducing training time by 33.6% and peak GPU memory usage by 89.4%, making it particularly attractive for resource-constrained applications.

4.3 Ablation Studies

We then evaluate PARSI through component ablations (Table 2) and comparisons of temporal range configurations (Figure 2). Based on these results, we make three observations: **(1) Temporal diversity is useful, but not uniformly beneficial.** Jointly modeling multiple ranges outperforms individual ranges on CESNET and ClusterTrace, while medium- or long-range branches alone perform slightly better on FNSPID. These results indicate that useful temporal information is distributed dif-

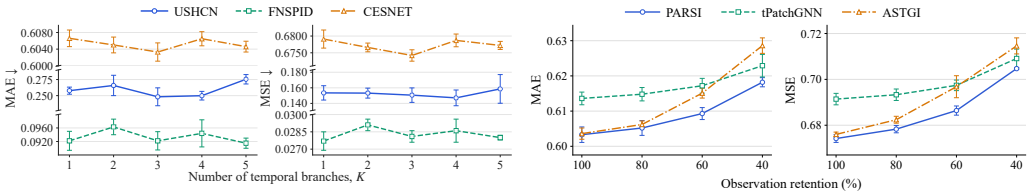

Figure 3: Model sensitivity and robustness analysis.

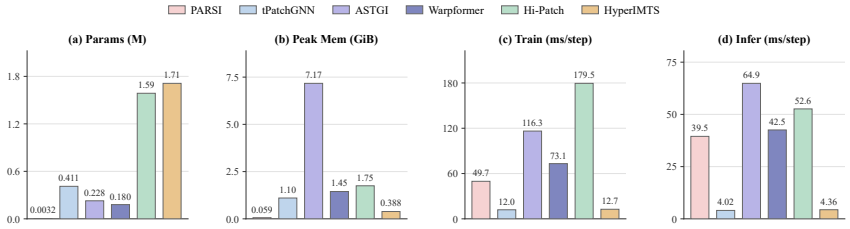

Figure 4: Comparison of computational efficiency and resource consumption.

ferently across datasets, motivating the use of parallel temporal ranges to accommodate diverse temporal requirements. **(2) Effective fusion matters beyond having multiple branches.** Equal-weight fusion retains all temporal branches but yields higher errors in every reported comparison, most notably on USHCN. This shows that access to multiple temporal ranges alone is insufficient, and adaptive fusion helps exploit their complementary information by adjusting their contributions. **(3) Cross-variable context complements temporal modeling selectively.** Removing inter-variable interaction has little effect on FNSPID, CESNET, and ClusterTrace, whereas USHCN shows a clearer loss, particularly in MAE. One possible explanation is that temporal histories already provide much of the useful predictive information on the former datasets, while other variables supply additional context on USHCN. The ablation therefore supports a dataset-dependent role for cross-variable modeling, rather than an equally large benefit across all settings.

4.4 Model Analysis

We further analyze PARSI under different settings. First, we study the effect of the number of temporal branches K in Figure 3. Performance remains relatively stable as K increases from 1 to 5, suggesting that additional branches do not necessarily improve forecasting accuracy. We then evaluate robustness by reducing the observation retention rate from 100% to 40%. As shown in Figure 3, PARSI consistently achieves the lowest MAE and MSE on CESNET, with performance degrading gradually as observations become sparser. Finally, we examine computational efficiency on USHCN illustrated in Figure 4. PARSI uses a small number of parameters as well as a GPU memory footprint while maintaining competitive forecasting accuracy, indicating that its performance does not rely on a large model size or high memory consumption.

5 Conclusion

PARSI highlights the importance of treating temporal range as a modeling choice rather than a fixed assumption in IMTS forecasting. Different datasets benefit from different historical contexts, and combining these contexts requires more than simply aggregating additional information. By separating temporal representation learning from adaptive integration, PARSI offers a flexible way to accommodate these differences while incorporating complementary cross-variable information. Its competitive forecasting accuracy, and robustness under increased observation sparsity and modest parameter and GPU memory requirements suggest that this flexibility need not come at a high computational cost. Overall, our findings provide a flexible modeling approach for real-world irregular time series applications such as healthcare monitoring, sensing, and system operation.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript writing and editing, including language refinement and sentence restructuring. They were also used for code-related assistance, including code formatting, debugging, readability improvement, and implementation refinement. AI tools additionally supported the organization and presentation of experimental results. The core research ideas, methodological design, experimental decisions, and final interpretations were developed and verified by the authors. All AI-assisted content and code were manually reviewed, and the authors take full responsibility for the final manuscript and results.

REPRODUCIBILITY STATEMENT

Implementation details and experimental settings are provided in Sections 3–4, with additional configurations and experimental details included in the supplementary material. The anonymous implementation of PARSI, together with dataset preparation instructions and scripts for reproducing the main experiments, is available at `https://anonymous.4open.science/r/PARSI-reproducibility-2CEA/`.

REFERENCES

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Siddhartha Bhandari, Neil Bergmann, Raja Jurdak, and Branislav Kusy. Time series analysis for spatial node selection in environment monitoring sensor networks. *Sensors*, 18(1):11, 2018. doi: 10.3390/s18010011.

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, pp. 21325–21337, 2021.

Ching Chang, Jeehyun Hwang, Yidan Shi, Haixin Wang, Wei Wang, Wen-Chih Peng, and Tien-Fu Chen. TIME-IMM: A dataset and benchmark for irregular multimodal multivariate time series. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

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.

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, pp. 7377–7388, 2019.

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*, pp. 4918–4927, 2024. doi: 10.1145/3637528.3671629.

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*, pp. 4353–4363. PMLR, 2020.

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*, pp. 35502–35518. PMLR, 2025.

Boyuan Li, Zhen Liu, Yicheng Luo, and Qianli Ma. Learning recursive multi-scale representations for irregular multivariate time series forecasting. In *International Conference on Learning Representations*, 2026.

Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 379 (2194):20200209, 2021. doi: 10.1098/rsta.2020.0209.

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 *International Conference on Learning Representations*, 2026a.

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, pp. 23873–23881, 2026b. doi: 10.1609/aaai.v40i28.39563.

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*, pp. 41494–41519. PMLR, 2025.

Giangiacomo Mercatali, Andre Freitas, and Jie Chen. Graph neural flows for unveiling systemic interactions among irregularly sampled time series. In *Advances in Neural Information Processing Systems*, volume 37, pp. 57183–57206, 2024.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *ICLR*, 2023.

Yulia Rubanova, Ricky T. Q. Chen, and David Duvenaud. Latent odes for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, volume 32, pp. 5320–5330, 2019.

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*, pp. 19388–19405. PMLR, 2022.

Satya Narayan Shukla and Benjamin M. Marlin. Interpolation-prediction networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2019.

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

Peng Wang, Longxi Feng, Wenhao Zhang, and Kanghua Hui. TWIST: An efficient spatial-temporal transformer with temporal window and sparse attention for traffic forecasting. *IEEE Internet of Things Journal*, 12(14):26799–26815, 2025. doi: 10.1109/JIOT.2025.3561542.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, et al. Timemixer: Decomposable multiscale mixing for time series forecasting. In *ICLR*, 2024.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

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, pp. 16255–16263, 2024. doi: 10.1609/aaai.v38i15.29560.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI*, 2023.

Jiawen Zhang, Shun Zheng, Wei Cao, Jiang Bian, and Jia Li. Warpformer: A multi-scale modeling approach for irregular clinical time series. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3273–3285. Association for Computing Machinery, 2023. doi: 10.1145/3580305.3599543.

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*, pp. 60179–60196. PMLR, 2024a. URL https://proceedings.mlr.press/v235/zhang24bw.html.

Weijia Zhang, Le Zhang, Jindong Han, Hao Liu, Yanjie Fu, Jingbo Zhou, Yu Mei, and Hui Xiong. Irregular traffic time series forecasting based on asynchronous spatio-temporal graph convolutional networks. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4302–4313, 2024b. doi: 10.1145/3637528.3671665.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325.

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, pp. 29115–29123, 2026. doi: 10.1609/aaai.v40i34.40149.