BEFORE RELATIONAL REASONING: RESTORING SYNCHRONICITY IN EVENT-CENTRIC IRREGULAR TIME SERIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Event-centric models for irregular multivariate time series commonly embed observations independently before learning their relations. This computation order leaves synchronous evidence outside the states from which relational reasoning begins, even though measurements recorded at the same timestamp jointly describe the underlying system. We identify this architectural mismatch as *pre-propagation synchronicity loss* and challenge the assumption that event representations should be constructed independently of available snapshot context. We formulate *pre-propagation synchronicity modeling* as a representation interface that separates context construction from relational forecasting. The interface reconstructs permutation-invariant snapshot context and fuses it with event evidence using local event-layout support. Across three clinical benchmarks and four forecasting architectures, the design improves mean MSE and MAE in all 12 model–dataset pairs. Component ablations support the contributions of synchronous context, adaptive fusion, and event-layout support. The consistent gains across attention-based, graph-based, and hypergraph-based forecasters establish the empirical reach of the principle beyond a single architecture. These results identify representation ordering as an overlooked design axis for irregular time-series forecasting.

1 INTRODUCTION

When should observations begin to interact? Event-centric irregular time-series models typically inherit a largely unexamined architectural assumption: construct each event state independently, then let a relational model integrate the evidence. Yet measurements recorded at the same timestamp already provide shared snapshot context. Under independent initialization, relational reasoning begins before that context has entered the event states. We identify this architectural mismatch as *pre-propagation synchronicity loss*. We challenge the assumption that sparse events should always be independently embedded before relational reasoning. This reframes synchronicity as a *representation-ordering* problem: retaining timestamp connections specifies where information can flow; constructing snapshot-aware states determines what information is available when reasoning begins.

Clinical records make this distinction concrete. A pulse measurement accompanied by blood pressure and oxygen saturation contributes to a patient-state snapshot whose meaning differs from an isolated pulse value. At the same time, observation patterns reflect clinical routines and measurement decisions: some variables are recorded frequently, while others are measured selectively. Thus, the input describes both a partially observed physiological process and the process that determines what is measured (Che et al., 2018; Tipirneni & Reddy, 2022). Clinical benchmarks including PhysioNet 2012, MIMIC-III, and MIMIC-IV expose these coupled sources of structure (Silva et al., 2012; Johnson et al., 2016; 2023). Throughout this paper, a snapshot means observations sharing a *recorded* time index under the benchmark's temporal resolution; these groups reflect the available temporal resolution, rather than exact physical simultaneity.

Event and set representations preserve sparse observations without dense value imputation (Horn et al., 2020; Tipirneni & Reddy, 2022). Graph and hypergraph forecasters additionally provide explicit paths for learning temporal and cross-variable dependencies (Yalavarthi et al., 2024; Li et al., 2025). These structures govern interactions among the resulting event representations.

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

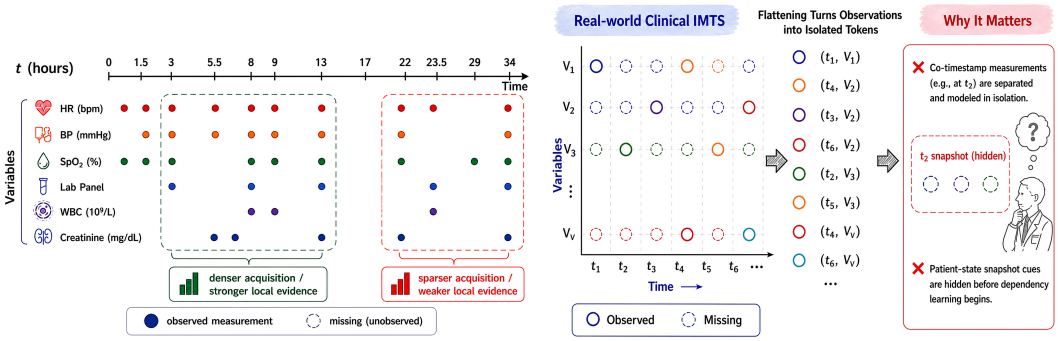

Figure 1: Clinical motivation for pre-propagation synchronicity modeling. Left: sparse records exhibit heterogeneous acquisition patterns. Right: independent event representations defer shared snapshot context to later relational reasoning. "Hidden" refers to context absent from initial event states; timestamp metadata remain available. Event-layout support summarizes local structural availability; it contains information from observed acquisition patterns and the predefined query coordinates that specify the forecasting layout of each episode.

At first glance, timestamp relations appear to resolve the synchronicity issue because co-recorded observations can exchange information during propagation. However, connectivity and state construction answer different questions. A timestamp edge determines which events may interact, but the first relational transformation is still computed from the states supplied to it. Under independent initialization, shared synchronous evidence enters those states only after relational computation has begun. This raises our central question: should relational models be asked to recover context that is already available when their input representations are constructed?

This is an *operation-ordering* question. Transforming isolated events and subsequently aggregating them need not produce the same representation as exposing their shared context before transformation. For a nonlinear transformation F and an aggregation operator A, in general $F(A(X)) \neq A(F(X))$. Section 3 makes this distinction precise with a counterexample. This minimal formal argument isolates the effect of computation order.

We formulate *pre-propagation synchronicity modeling* as an architecture-agnostic representation interface. The interface separates representation construction from relational forecasting: $\mathcal{S}$ first reconstructs shared timestamp context and conditions its interaction with event evidence on local event-layout support; a forecaster then reasons from these context-aware states. The information path becomes *event representation, synchronicity restoration, and relational reasoning*. Across four forecasting architectures and three clinical datasets, this design improves both mean MSE and MAE in all 12 model–dataset pairs. We call the hypergraph realization **MissHyper**; it achieves the lowest errors among all evaluated forecasting configurations.

Our contributions cover problem formulation, interface design, and empirical evaluation:

- We expose an overlooked architectural mismatch in event-centric irregular time-series modeling, which we term pre-propagation synchronicity loss: relational reasoning begins from independently constructed event states even when synchronous evidence is already available. This identifies context timing as a representation-learning question.
- We formulate synchronicity restoration as an architecture-agnostic representation interface before relational forecasting. The realization reconstructs permutation-invariant snapshot context and fuses it with event evidence using event-layout support.
- Applying the proposed interface to four forecasting architectures across three clinical benchmarks reduces both mean MSE and MAE in all 12 model–dataset combinations. Ablations of MissHyper further isolate the contributions of synchronous context, adaptive fusion, and the inclusion of local event-layout support in event representations.

2 Related Work

Representing irregular observations. Irregular time-series models differ in how they expose time, missingness, and values to a learner. Interpolation-based methods align sparse observations with a reference representation (Shukla & Marlin, 2019; 2021); recurrent models incorporate elapsed time and missingness into state updates (Neil et al., 2016; Baytas et al., 2017; Che et al., 2018). Continuous-time models instead evolve latent states between observations through differential equations or flows (Rubanova et al., 2019; De Brouwer et al., 2019; Kidger et al., 2020; Biloš et al., 2021; Schirmer et al., 2022). Set and event models preserve individual observations as sparse inputs (Horn et al., 2020; Tipirneni & Reddy, 2022; Chowdhury et al., 2023). Our focus is the construction of the states supplied to relational learning: whether co-recorded observations contribute shared context before those states undergo relational transformations. This question concerns models with independently formed event states, rather than every irregular time-series architecture.

Relational structure and the order of computation. Graph models make dependencies among variables and observations explicit, as in Raindrop and GraFITi (Zhang et al., 2022; Yalavarthi et al., 2024). Patching and hierarchical methods introduce additional temporal structure before or during graph reasoning (Zhang et al., 2024; Luo et al., 2025). Hypergraph learning represents multiway relations with shared hyperedges (Feng et al., 2019; Bai et al., 2021); HyperIMTS uses time and variable hyperedges for irregular forecasting (Li et al., 2025). These relational structures are compatible with different input representations. Our work studies when shared snapshot evidence enters event representations, complementing the choice of relational structure. The model-paired evaluation tests this design across attention-based, graph-based, and hypergraph-based forecasters.

Missingness as measurement-process information. Missingness-aware learning uses observation patterns as predictive inputs rather than treating all missing values as interchangeable (Che et al., 2018; Tipirneni & Reddy, 2022). Such patterns can depend on both the underlying state and the acquisition procedure (Rubin, 1976; Little & Rubin, 2002). Our framework uses local event-layout support to condition how events contribute to, and borrow from, synchronous context. This differs in emphasis from imputing unobserved values (Cao et al., 2018; Tashiro et al., 2021; Du et al., 2023): the operator reconstructs a representation of *available* co-timestamp evidence. This support signal contains measurement-process information from observed event positions and also reflects predefined query coordinates; it is not a calibrated uncertainty estimate.

3 Pre-Propagation Synchronicity Modeling

The architectural mismatch. Let $\mathcal{O}_t = \{n : \tau_n = t,\ n \text{ is observed}\}$ be the set of events at a recorded timestamp. An event-independent encoder constructs $\mathbf{h}_n$ from local inputs, so changing another observed value in $\mathcal{O}_t$ cannot change $\mathbf{h}_n$ before propagation. We call the absence of this within-snapshot dependence *pre-propagation synchronicity loss*. Timestamp metadata remain available; the loss concerns shared evidence absent from event states at the boundary where relational reasoning begins. Restoring synchronicity means making $\mathbf{z}_n$ depend on observed co-timestamp evidence before the first relational transformation.

Why a timestamp relation does not settle the ordering question. Relational computations depend on the states from which their messages are formed. For example, node-to-hyperedge attention constructs keys and values from event and incident hyperedge states (Li et al., 2025). Exposing shared context in the event states therefore changes the inputs to the first attention update even when the incidence structure is identical. For a set H of event embeddings, the following expressions contrast independent transformation with context-first computation:

$$\underbrace{H \xrightarrow{F} F(H) \xrightarrow{A} A(F(H))}_{\text{transform independent states, then aggregate}}, \qquad \underbrace{H \xrightarrow{A} A(H) \xrightarrow{F} F(A(H))}_{\text{expose context before transformation}}. \tag{1}$$

Here $F(H)$ applies F to each event. These expressions isolate an ordering issue, rather than reproduce an entire hypergraph layer. Our framework retains individual event states and fuses them with a shared summary to provide context for the first relational transformation.

Proposition 1 (Non-commutativity of aggregation and transformation)**.** *Let A be arithmetic-mean aggregation of scalar event states. There exists a nonlinear event transformation F and a two-event snapshot H such that $F(A(H)) \neq A(F(H))$.*

Proof. Take $H = \{-1, 1\}$ and $F(x) = \max(0, x)$. Then $F(A(H)) = F(0) = 0$, whereas $A(F(H)) = (0 + 1)/2 = 1/2$. □

Proposition 1 shows that exposing synchronous context before nonlinear transformation and aggregating transformed events are not generally interchangeable. This minimal argument formalizes the computation-order distinction underlying the representation interface.

Why the distinction matters. Relational architectures specify how information moves through attention, graph edges, hyperedges, or temporal patches. Pre-propagation synchronicity addresses the information content of the states on which those operators act. Two models can retain exactly the same relational structure while exposing different evidence to their first transformation. The distinction is therefore between connectivity and representation construction: connectivity determines which observations may interact, while representation ordering determines what evidence is available when that interaction begins. Context restoration complements relational architecture design by constructing inputs from evidence already present in the event layout. It motivates treating representation construction and relational reasoning as jointly designed stages, instead of assuming that subsequent interactions will always compensate for independently initialized event states.

An architecture-agnostic representation interface. We express this principle as

$$Z_0 = \mathcal{S}_\eta(X, \Omega), \qquad \widehat{Y} = \mathcal{B}_\theta(Z_0, \mathcal{R}), \tag{2}$$

where X contains observed values, event identities, and query placeholders; Ω describes known event positions; and $\mathcal{R}$ contains the forecaster's relational structure. $\mathcal{S}_\eta$ reconstructs observable snapshot context and conditions its use on event-layout support. $\mathcal{B}_\theta$ then learns relational dependencies from these states. The interface separates representation construction from relational forecasting. Section 5.2 evaluates the proposed design across four forecasting architectures.

4 Event-Layout-Aware Synchronicity Restoration

The interface in Equation 2 defines two jointly designed stages: pre-propagation representation construction and relational forecasting. Our framework implements $\mathcal{S}_\eta$ through a permutation-invariant reconstruction of co-timestamp evidence and event-layout-conditioned fusion. The operator preserves sparse event modeling: it does not impute unmeasured values or require a dense value grid. Figure 2 illustrates the information flow using the hypergraph realization, MissHyper.

4.1 Sparse events and event-layout support

For episode b, let $\mathcal{O}_b$ and $\mathcal{Q}_b$ denote observed and query events, and let $\mathcal{V}_b = \mathcal{O}_b \cup \mathcal{Q}_b$. Each observed event has timestamp $\tau_{b,n}$, variable identity $u_{b,n}$, and value $x_{b,n}$. Queries have known coordinates and hidden values. Define

$$\tilde{x}_{b,n} = \begin{cases} x_{b,n}, & n \in \mathcal{O}_b, \\ 0, & n \in \mathcal{Q}_b, \end{cases} \qquad q_{b,n} = \mathbf{1}[n \in \mathcal{Q}_b]. \tag{3}$$

The role flag distinguishes a query placeholder from a measured zero. Time and variable identities define the event layout and the downstream incidence relations.

Let $\Omega_b \in \{0, 1\}^{L \times C}$ indicate the known event positions: both observed inputs and predefined query coordinates. Using a clipped local window $\mathcal{W}_w(\ell)$, define event-layout support as

$$\rho_{b,\ell,u} = \frac{1}{|\mathcal{W}_w(\ell)|} \sum_{s \in \mathcal{W}_w(\ell)} \Omega_{b,s,u}, \qquad \rho_{b,n} = \rho_{b,\tau_{b,n},u_{b,n}} \in [0, 1]. \tag{4}$$

Thus ρ measures local *event-layout support*. It contains measurement-process information through observed event positions and prediction-layout information through predefined query coordinates. The task supplies query coordinates independently of target values, which never enter this mask.

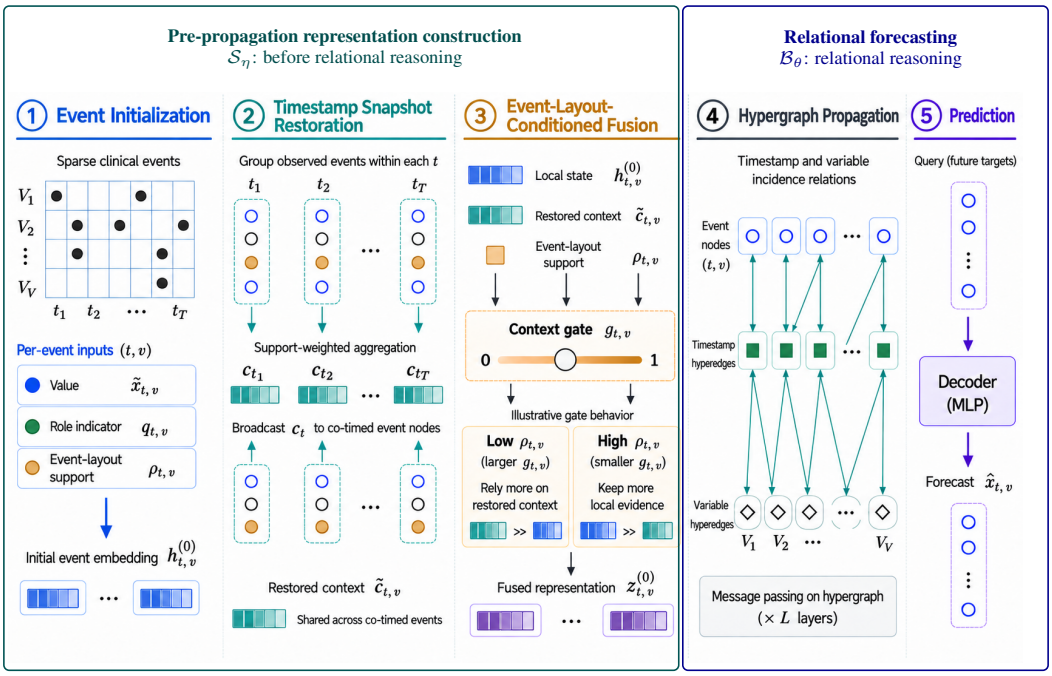

Figure 2: The representation interface $\mathcal{S}_\eta \rightarrow \mathcal{B}_\theta$, illustrated by MissHyper. Stages 1–3 construct snapshot-aware event states before relational reasoning; stages 4–5 forecast from those states. Only observed values contribute to snapshot context; known query positions can contribute to the structural support cue. The gate examples are illustrative; responses are learned.

The event states supplied to snapshot reconstruction are

$$\mathbf{s}_{b,n} = [\tilde{x}_{b,n}, q_{b,n}, \rho_{b,n}]^\top, \qquad \mathbf{h}^{(0)}_{b,n} = \phi_{\text{in}}(\mathbf{s}_{b,n}) \in \mathbb{R}^d. \tag{5}$$

The window size w sets the support scale and is selected on validation data. Support distinguishes event layouts that share the same measured value. Its predictive role is learned from the forecasting objective; it is not assigned a fixed interpretation as measurement accuracy.

4.2 Permutation-Invariant Snapshot Reconstruction

At each timestamp ℓ, the operator reconstructs shared context from the *observed* events:

$$\mathbf{c}_{b,\ell} = \frac{\sum_{n \in \mathcal{O}_b : \tau_{b,n} = \ell} \rho_{b,n} \mathbf{h}^{(0)}_{b,n}}{\sum_{n \in \mathcal{O}_b : \tau_{b,n} = \ell} \rho_{b,n} + \epsilon}, \qquad \tilde{\mathbf{c}}_{b,n} = \phi_c(\mathbf{c}_{b,\tau_{b,n}}). \tag{6}$$

Support weighting combines instantaneous co-occurrence with nearby event-layout information. The sums make the summary invariant to the order of events within a timestamp. The projection ϕ_c maps that shared summary to the event embedding space. Reconstruction produces context from available evidence and does not estimate unobserved measurements or a complete patient state.

For a timestamp with no observations, the numerator is zero and $\epsilon > 0$ makes the context well-defined. A query-only timestamp therefore receives $\phi_c(\mathbf{0})$, which may be nonzero if the projection has a bias. It receives no observed snapshot evidence at that timestamp. In prefix forecasting, the restored context primarily enriches historical observations, from which future query nodes obtain information through subsequent propagation. Query targets never contribute to the summary.

4.3 Event-Layout-Conditioned Fusion

A shared summary need not be equally useful to every event. The fusion operator learns how much of it to expose in each event state using the event content, the reconstructed

context, and the support signal. Let $\mathbf{H}_b^{(0)}$, $\widetilde{\mathbf{C}}_b$, and $\boldsymbol{\rho}_b$ stack these quantities over nodes. The dimension-wise context gate and restored states are

$$\mathbf{A}_b = \sigma\Big(\phi_g\Big([\mathbf{H}_b^{(0)} \,\|\, \widetilde{\mathbf{C}}_b \,\|\, \boldsymbol{\rho}_b]\Big)\Big), \tag{7}$$

$$\mathbf{Z}_b^{(0)} = \mathbf{H}_b^{(0)} + \mathbf{A}_b \odot (\widetilde{\mathbf{C}}_b - \mathbf{H}_b^{(0)}). \tag{8}$$

Here σ is the sigmoid, $\|$ denotes concatenation, and $\mathbf{A}_b \in (0,1)^{|\mathcal{V}_b| \times d}$. A value near zero preserves the event coordinate; a value near one moves it toward the corresponding context coordinate. This interpolation lets the first relational transformation act on both event-specific and shared evidence. The gate learns its response to support jointly with forecasting.

4.4 Relational forecasting

For the hypergraph realization, we use timestamp-node and variable-node incidence relations $\mathcal{A}_b^t$ and $\mathcal{A}_b^u$ for relational propagation (Li et al., 2025):

$$(\mathbf{Z}_b^{(K)}, \mathbf{T}_b^{(K)}, \mathbf{U}_b^{(K)}) = \mathcal{B}_\theta(\mathbf{Z}_b^{(0)}, \mathcal{A}_b^t, \mathcal{A}_b^u). \tag{9}$$

$\mathbf{T}_b^{(K)}$ and $\mathbf{U}_b^{(K)}$ are the final timestamp and variable hyperedge states. Timestamp relations connect co-recorded events, while variable relations connect observations of the same channel across time. Propagation learns these dependencies from states that already expose shared snapshot context.

For a query node, the decoder and forecasting objective are

$$\hat{x}_{b,n} = \psi_o\Big([\mathbf{z}_{b,n}^{(K)} \,\|\, \mathbf{t}_{b,\tau_{b,n}}^{(K)} \,\|\, \mathbf{u}_{b,u_{b,n}}^{(K)}]\Big), \tag{10}$$

$$\mathcal{L} = \mathbb{E}_{(b,n)\in\mathcal{Q}}\left[\|\hat{x}_{b,n} - x_{b,n}\|_2^2\right]. \tag{11}$$

Observed nodes provide evidence but do not contribute to the query forecasting loss. At fixed embedding dimension, snapshot aggregation and fusion scale linearly with the number of events. Window-based support gathering costs $O(|\mathcal{V}_b|w)$ with indexed mask access. This step constructs the states supplied to propagation without adding hyperedges or iterative propagation.

5 Experiments

Across four architectures and three clinical benchmarks, all 12 model–dataset pairs improve in both mean MSE and MAE. We organize the evaluation around three questions: whether pre-propagation synchronicity modeling transfers across forecasting architectures, how its hypergraph realization compares with existing clinical forecasters, and how synchronous context and event-layout conditioning contribute to the resulting representations.

5.1 Evaluation protocol

We evaluate on PhysioNet 2012 (P12), MIMIC-III, and MIMIC-IV, using recorded time resolutions of 1 hour, 30 minutes, and 1 minute, respectively (Silva et al., 2012; Johnson et al., 2016; 2023). The datasets contain 36, 96, and 100 variables. Episodes are split 80%/10%/10% into training, validation, and test sets. Each model observes a prefix and predicts values at predefined query positions. For the benchmark comparison and component ablations, we report MSE and MAE as mean $\pm$ standard deviation over seeds 2024–2028. Validation data select the checkpoint and the support window; test data are held out from selection. Dataset statistics, baseline families, and optimization settings for the benchmark evaluation are detailed in Appendix A.

We evaluate the design with tPatchGNN, Warpformer, GraFITi, and HyperIMTS (Zhang et al., 2024; 2023; Yalavarthi et al., 2024; Li et al., 2025). Each model is compared with its counterpart using our design. The hypergraph configuration, denoted MissHyper, retains the original incidence structure, propagation operators, decoder, and propagation hyperparameters; its representation parameters are learned jointly with the forecaster under the query forecasting objective.

Table 1: Pre-propagation synchronicity modeling across four forecasting models (mean ± std). Each pair compares the original model with the same model using our design. Bold marks the lower mean within each pair. HyperIMTS + our design is denoted MissHyper elsewhere. Lower is better.

	P12		MIMIC-III		MIMIC-IV	
Model	MSE	MAE	MSE	MAE	MSE	MAE
tPatchGNN	0.3168 ± 0.0049	0.3715 ± 0.0046	0.4465 ± 0.0107	0.4091 ± 0.0083	0.2761 ± 0.0019	0.3115 ± 0.0014
+ our design	**0.3143** ± 0.0043	**0.3656** ± 0.0034	**0.4429** ± 0.0094	**0.4022** ± 0.0081	**0.2693** ± 0.0004	**0.3075** ± 0.0009
Warpformer	0.3089 ± 0.0014	0.3649 ± 0.0018	0.4326 ± 0.0031	0.4047 ± 0.0016	0.2728 ± 0.0025	0.3129 ± 0.0020
+ our design	**0.3031** ± 0.0012	**0.3612** ± 0.0012	**0.4266** ± 0.0017	**0.3973** ± 0.0012	**0.2683** ± 0.0022	**0.3096** ± 0.0009
GraFITi	0.3098 ± 0.0017	0.3651 ± 0.0031	0.4374 ± 0.0412	0.4148 ± 0.0285	0.2467 ± 0.0008	0.3018 ± 0.0005
+ our design	**0.3020** ± 0.0009	**0.3625** ± 0.0027	**0.4303** ± 0.0399	**0.4088** ± 0.0270	**0.2415** ± 0.0001	**0.2975** ± 0.0001
HyperIMTS	0.3010 ± 0.0027	0.3619 ± 0.0045	0.4009 ± 0.0122	0.3745 ± 0.0099	0.2147 ± 0.0020	0.2793 ± 0.0020
+ our design	**0.2962** ± 0.0008	**0.3561** ± 0.0011	**0.3860** ± 0.0033	**0.3643** ± 0.0018	**0.2081** ± 0.0025	**0.2751** ± 0.0011

Table 2: Multi-step forecasting performance (mean ± std over 5 runs). Lower is better.

	P12 (PhysioNet)		MIMIC-III		MIMIC-IV	
Method	MSE	MAE	MSE	MAE	MSE	MAE
PrimeNet	0.7981 ± 0.0001	0.6872 ± 0.0002	0.9105 ± 0.0002	0.6657 ± 0.0001	0.6682 ± 0.0001	0.5891 ± 0.0002
NeuralFlows	0.4098 ± 0.0031	0.4492 ± 0.0024	0.6127 ± 0.0098	0.5342 ± 0.0061	0.4712 ± 0.0021	0.4782 ± 0.0012
CRU	0.6215 ± 0.0042	0.5791 ± 0.0035	0.5921 ± 0.0101	0.5178 ± 0.0050	0.4368 ± 0.0019	0.4574 ± 0.0009
mTAN	0.3826 ± 0.0040	0.4318 ± 0.0031	0.9473 ± 0.1092	0.6811 ± 0.0472	0.5361 ± 0.0118	0.5203 ± 0.0061
SeFT	0.7694 ± 0.0020	0.6728 ± 0.0030	0.9286 ± 0.0017	0.6641 ± 0.0009	0.6735 ± 0.0003	0.5912 ± 0.0004
GNeuralFlow	0.8251 ± 0.0287	0.6812 ± 0.0094	0.9012 ± 0.0186	0.6483 ± 0.0069	0.5023 ± 0.0017	0.4915 ± 0.0005
GRU-D	0.3467 ± 0.0025	0.4010 ± 0.0016	0.4817 ± 0.0094	0.4563 ± 0.0053	0.6598 ± 0.0024	0.5834 ± 0.0015
Raindrop	0.3519 ± 0.0023	0.4087 ± 0.0021	0.6782 ± 0.1714	0.5528 ± 0.0821	0.3429 ± 0.0038	0.3896 ± 0.0028
Warpformer	0.3089 ± 0.0014	0.3649 ± 0.0018	0.4326 ± 0.0031	0.4047 ± 0.0016	0.2728 ± 0.0025	0.3129 ± 0.0020
tPatchGNN	0.3168 ± 0.0049	0.3715 ± 0.0046	0.4465 ± 0.0107	0.4091 ± 0.0083	0.2761 ± 0.0019	0.3115 ± 0.0014
GraFITi	0.3098 ± 0.0017	0.3651 ± 0.0031	0.4374 ± 0.0412	0.4148 ± 0.0285	0.2467 ± 0.0008	0.3018 ± 0.0005
HyperIMTS	0.3010 ± 0.0027	0.3619 ± 0.0045	0.4009 ± 0.0122	0.3745 ± 0.0099	0.2147 ± 0.0020	0.2793 ± 0.0020
MissHyper	**0.2962** ± 0.0008	**0.3561** ± 0.0011	**0.3860** ± 0.0033	**0.3643** ± 0.0018	**0.2081** ± 0.0025	**0.2751** ± 0.0011

5.2 Testing the principle across forecasting architectures

Table 1 places each original forecaster directly above its counterpart using our design. This is the central test of the representation principle: whether changing the information available before relational reasoning benefits different forecasting architectures. Mean MSE and MAE decrease for every model on every dataset: all 12 model–dataset pairs improve on both metrics.

Beyond the hypergraph realization, relative MSE reductions range from 0.79% to 2.52%. On P12, for example, MSE decreases from 0.3168 to 0.3143 for tPatchGNN, from 0.3089 to 0.3031 for Warpformer, and from 0.3098 to 0.3020 for GraFITi. The corresponding improvements on MIMIC-IV are 2.46%, 1.65%, and 2.11%. The hypergraph realization, MissHyper, gives the lowest mean errors overall, while the other three pairs show that the benefit extends to the additional forecasting architectures included in this evaluation.

The consistent direction of improvement across these architectures supports representation construction as a shared design axis. Their inductive biases differ: tPatchGNN combines transformable temporal patches with graph reasoning, Warpformer combines multi-scale warping with attention, GraFITi formulates forecasting through a sparse bipartite graph, and the hypergraph model uses multiway propagation (Zhang et al., 2024; 2023; Yalavarthi et al., 2024; Li et al., 2025). Gains across these configurations suggest that the effect is not tied to one relational topology or interaction operator. This is consistent with our hypothesis that context-independent event states can create a shared representational bottleneck across otherwise different forecasters. The benchmark comparison and ablations next examine the accuracy and constituent choices of the hypergraph realization.

5.3 Clinical forecasting performance

Table 2 situates MissHyper among existing forecasters. Relative to the direct hypergraph baseline, mean MSE falls by 1.59% on P12, 3.72% on MIMIC-III, and 3.07% on MIMIC-IV; the correspond-

Table 3: Ablation study on multi-step forecasting (mean ± std over 5 runs). Lower is better.

	P12 (PhysioNet)		MIMIC-III		MIMIC-IV	
Variant	MSE	MAE	MSE	MAE	MSE	MAE
MissHyper	**0.2962** ± 0.0008	**0.3561** ± 0.0011	**0.3860** ± 0.0033	**0.3643** ± 0.0018	**0.2081** ± 0.0025	**0.2751** ± 0.0011
w/o Snapshot	0.2984 ± 0.0012	0.3587 ± 0.0015	0.3942 ± 0.0047	0.3710 ± 0.0026	0.2110 ± 0.0029	0.2783 ± 0.0016
w/o Gate	0.2976 ± 0.0010	0.3579 ± 0.0013	0.3909 ± 0.0039	0.3676 ± 0.0022	0.2098 ± 0.0026	0.2766 ± 0.0014
w/o Support	0.2981 ± 0.0013	0.3584 ± 0.0016	0.3924 ± 0.0043	0.3691 ± 0.0025	0.2105 ± 0.0028	0.2774 ± 0.0015

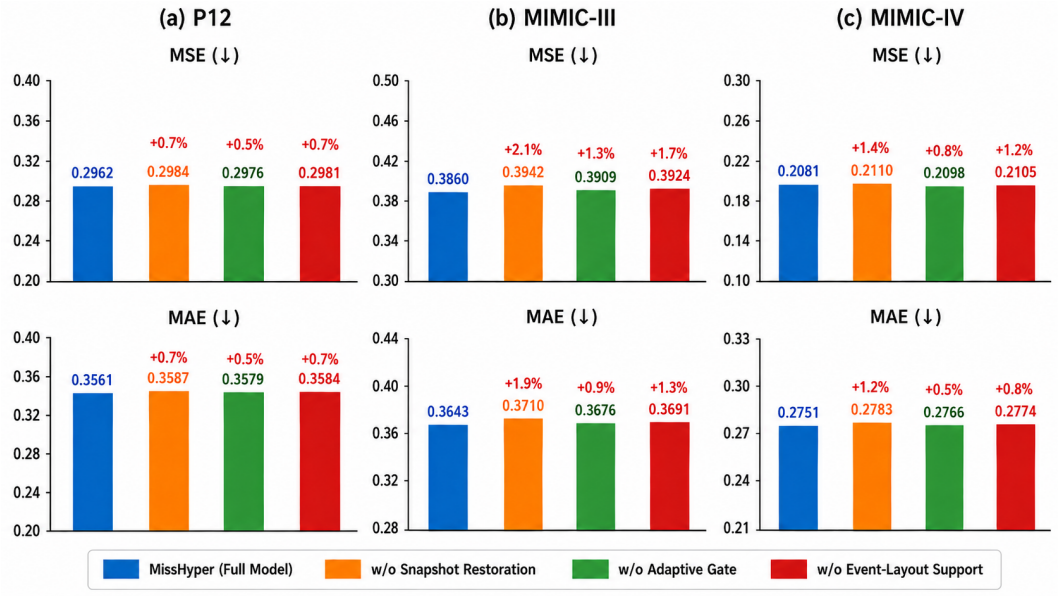

Figure 3: Component ablation results on the three clinical benchmarks. Values correspond to the means in Table 3; the table also reports run-to-run standard deviations. Removing synchronous context has the largest mean-error effect among the reported removals.

ing MAE reductions are 1.60%, 2.72%, and 1.50%. MissHyper achieves the lowest mean errors among the listed methods across both metrics and all three datasets.

MIMIC-III shows the strongest relative gain and a reduction in MSE standard deviation from 0.0122 to 0.0033. It also has more variables and fewer average observations per episode than P12 (Appendix A), motivating the event-layout interpretation discussed in Section 5.6.

5.4 Context and Event-Layout Conditioning

Table 3 and Figure 3 isolate snapshot restoration, adaptive gating, and event-layout support within the same architecture and training protocol. Every removal increases both mean MSE and MAE on all datasets. Removing snapshot restoration gives the largest degradation; on MIMIC-III, MSE rises from 0.3860 to 0.3942. Removing the gate or support gives 0.3909 or 0.3924, respectively. These results support incorporating co-timestamp evidence before relational forecasting and conditioning its use on event content and local event-layout support.

5.5 Support Scale and Interpretation

The support window is selected on validation data from $w \in \{2, 3, \ldots, 14\}$, giving $w = 3$ for P12, $w = 7$ for MIMIC-III, and $w = 6$ for MIMIC-IV. These settings are fixed for the main test evaluation. Since time resolutions differ, the same numerical window would represent different physical durations across datasets. Support should therefore be interpreted in terms of the recorded temporal resolution used to construct each dataset's event layout.

The window-sensitivity visualization is provided in Appendix B. The selected settings and their reported test means are those in Table 2. The dataset-specific choices allow the representation to use an appropriate local scale for event-layout support.

5.6 Design implications

Representation ordering and relational capacity concern different stages of forecasting. Increasing depth or changing an interaction operator modifies how a forecaster processes its input states. Our interface changes what evidence those states contain before the first relational computation. A relational model may recover snapshot context through later interactions, but this defers the use of information that was already available at input time. Exposing it at the representation boundary instead allows the first learned interaction to depend on both an event and its co-recorded context. The distinction concerns the inputs to relational transformations, rather than an additional scalar feature attached to an otherwise unchanged event: each restored state can depend on its own value and the available values of other co-timestamp observations.

Irregular clinical data make this distinction concrete because measurements are sparse, asynchronous, and jointly informative. Observations sharing a recorded timestamp can describe complementary aspects of patient state, while the local event layout records the availability of that evidence. Support conditions how events contribute to and borrow from the shared context; it includes observed positions and predefined query coordinates. The larger relative improvement on MIMIC-III is compatible with the relevance of context under heterogeneous event layouts. Dataset differences, however, do not isolate sparsity as the cause of that gain. The ablations complement this comparison by isolating context restoration and adaptive fusion within MissHyper.

More broadly, event-centric forecasting couples two design decisions: what evidence enters each event representation, and how the resulting states interact. The four-model evaluation indicates that representation construction deserves explicit attention alongside the choice of attention or graph structure. Pre-propagation synchronicity is one concrete way to design this boundary using relations already observable in the input. Applying the perspective to another setting requires specifying its meaningful event groups, available coordinates, and prediction protocol. Event initialization is a modeling decision that shapes the evidence available to relational reasoning.

6 Conclusion

We introduced pre-propagation synchronicity loss as an architectural mismatch in event-centric irregular time-series representation construction: synchronous evidence is available, yet relational reasoning begins from independently constructed event states. Our framework changes the information available at this computation boundary through an event-layout-aware representation interface. Across four distinct forecasting architectures and three clinical benchmarks, all 12 model–dataset pairs improve in both mean MSE and MAE. MissHyper, the hypergraph realization, achieves the lowest errors among the evaluated configurations, while component ablations support the contributions of context restoration and adaptive fusion. These findings establish pre-propagation representation construction as a practical and transferable design principle across the evaluated architectures. The same principle improves attention-based, graph-based, and hypergraph-based forecasters by changing the evidence available before relational reasoning. Our results suggest that deciding when observations interact can be as consequential as deciding how they interact.

7 Limitations

The empirical evaluation covers four forecasting architectures and three clinical datasets; extending the analysis to other domains and observation regimes is an important direction for future work. Synchronicity is defined at the recorded temporal resolution, which can group measurements acquired at different physical times. The support signal depends on the known event layout, including predefined query coordinates, and a validation-selected local window. A more detailed analysis of learned gating behavior may further clarify the mechanism.

AI USE STATEMENT

Generative AI tools were used for language editing and stylistic refinement of the manuscript. All scientific content, including the methodology, experiments, analyses, and conclusions, was developed and verified by the authors. The authors take full responsibility for the final manuscript.

ETHICS STATEMENT

This work studies forecasting on clinical benchmarks. Forecasting errors, dataset bias, and differences in acquisition practices can affect clinical applicability. The method and the reported benchmark results do not establish safety for clinical deployment. Clinical use would require external validation, calibration assessment, privacy-preserving data handling, and subgroup evaluation.

REPRODUCIBILITY STATEMENT

Section 4 specifies the representation operator, the treatment of query values and coordinates, and the forecasting objective. Section 5.1 and Appendix A describe the dataset split, reported random seeds, optimization settings, and validation-based window selection. Proposition 1 includes the complete proof of its illustrative ordering claim.

REFERENCES

Song Bai, Feihu Zhang, and Philip H. S. Torr. Hypergraph convolution and hypergraph attention. *Pattern Recognition*, 110:107637, 2021. doi: 10.1016/j.patcog.2020.107637.

Inci M. Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K. Jain, and Jiayu Zhou. Patient subtyping via time-aware LSTM networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 65–74, 2017. doi: 10.1145/3097983.3097997.

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.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. BRITS: Bidirectional recurrent imputation for time series. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8:6085, 2018. doi: 10.1038/s41598-018-24271-9.

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, pp. 7184–7192, 2023. doi: 10.1609/aaai.v37i6.25876.

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.

Wenjie Du, David Côté, and Yan Liu. SAITS: Self-attention-based imputation for time series. *Expert Systems with Applications*, 219:119619, 2023. doi: 10.1016/j.eswa.2023.119619.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3558–3565, 2019. doi: 10.1609/aaai.v33i01.33013558.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6:96, 2019. doi: 10.1038/s41597-019-0103-9.

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, 2020.

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. doi: 10.1038/sdata.2016.35.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023. doi: 10.1038/s41597-022-01899-x.

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.

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.

Roderick J. A. Little and Donald B. Rubin. *Statistical Analysis with Missing Data*. Wiley, 2 edition, 2002. doi: 10.1002/9781119013563.

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*, 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. Curran Associates, Inc., 2024. doi: 10.52202/079017-1822.

Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. Phased LSTM: Accelerating recurrent network training for long or event-based sequences. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Sanjay Purushotham, Chuizheng Meng, Zhengping Che, and Yan Liu. Benchmarking deep learning models on large healthcare datasets. *Journal of Biomedical Informatics*, 83:112–134, 2018. doi: 10.1016/j.jbi.2018.04.007.

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

Donald B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976. doi: 10.1093/biomet/63.3.581.

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, 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.

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 *2012 Computing in Cardiology*, pp. 245–248, 2012.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 24804–24816, 2021.

Sindhu Tipirneni and Chandan K. Reddy. Self-supervised transformer for sparse and irregularly sampled multivariate clinical time-series. *ACM Transactions on Knowledge Discovery from Data*, 16(6):105:1–105:17, 2022. doi: 10.1145/3516367.

Shirly Wang, Matthew B. A. McDermott, Geeticka Chauhan, Michael C. Hughes, Tristan Naumann, and Marzyeh Ghassemi. MIMIC-Extract: A data extraction, preprocessing, and representation pipeline for MIMIC-III. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 222–235, 2020. doi: 10.1145/3368555.3384469.

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.

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, 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, 2024.

Xiang Zhang, Marko Zeman, Theodoros Tsiligkaridis, and Marinka Zitnik. Graph-guided network for irregularly sampled multivariate time series. In *International Conference on Learning Representations*, 2022.

Table 4: Dataset statistics for clinical IMTS forecasting. Avg. Obs. denotes the average number of observations per episode in the corresponding clinical dataset.

Dataset	Num. Vars	Num. Samples	Avg. Obs.	Max Length
P12	36	11,981	308.6	47
MIMIC-III	96	21,250	144.6	96
MIMIC-IV	100	17,874	304.8	971

A Dataset and Training Details

Datasets. We use three clinical IMTS benchmarks: PhysioNet 2012 (P12), MIMIC-III, and MIMIC-IV (Silva et al., 2012; Johnson et al., 2016; 2023). PhysioNet 2012 provides ICU time-series records from the first 48 hours of patient stays and is processed at a 1-hour resolution. MIMIC-III is a large critical-care EHR benchmark whose ICU measurements are discretized into 30-minute intervals under the same setup. MIMIC-IV extends the MIMIC data ecosystem with more recent and denser ICU records, which are processed at a 1-minute resolution. Because clinical benchmark construction and extraction choices can affect time-series evaluation, we follow established preprocessing protocols for these datasets (Purushotham et al., 2018; Harutyunyan et al., 2019; Wang et al., 2020). These datasets contain sparse and asynchronous ICU measurements with substantial variable-wise differences in recording frequency. As shown in Table 4, the datasets differ in scale, variable dimensionality, observation density, and sequence length. This diversity allows us to test MissHyper across different sparsity and asynchrony regimes.

Protocol. Each patient episode is represented as sparse timestamp-variable records rather than a dense time-by-variable grid. The model receives an observed prefix and predicts values at query positions within the forecasting horizon. Following prior clinical IMTS forecasting protocols (Li et al., 2025), we split episodes into training, validation, and test sets with an 80%/10%/10% ratio. Models are trained with shuffled mini-batches from the training split. The validation split is used only for early stopping and event-layout-support window selection. For each seed, the checkpoint selected on the validation split is evaluated on the held-out test split. We use the dataset-specific discretization described above, following prior IMTS forecasting protocols.

Baselines and implementation. We compare MissHyper with representative methods for irregular multivariate time-series forecasting. Event- and attention-based baselines include PrimeNet, mTAN, and SeFT (Chowdhury et al., 2023; Shukla & Marlin, 2021; Horn et al., 2020). Continuous-time and neural-flow baselines include NeuralFlows, CRU, and GNeuralFlow (Biloš et al., 2021; Schirmer et al., 2022; Mercatali et al., 2024). GRU-D represents missingness-aware recurrent modeling (Che et al., 2018). Graph-structured and recent irregular forecasting baselines include Raindrop, Warpformer, tPatchGNN, and GraFITi (Zhang et al., 2022; 2023; 2024; Yalavarthi et al., 2024). The most controlled comparison is the direct hypergraph baseline, where MissHyper uses the same construction and propagation operators but replaces the event initialization stage (Li et al., 2025). All experiments are conducted on a Linux server with a single NVIDIA GeForce RTX 4090 GPU. The implementation uses PyTorch 2.9.1. We optimize all trainable parameters with Adam. The learning rate is kept at L_0 for the first three epochs and then decayed as $L_n = L_0 \times 0.8^{n-3}$ for epoch $n > 3$. Models are trained for up to 300 epochs with an early-stopping patience of 10 epochs. We report MSE and MAE as mean $\pm$ standard deviation over five random seeds, 2024–2028. For MissHyper, the batch size is 32 and the initial learning rate is 5×10^{-4}. Propagation-related hyperparameters are kept fixed across the direct hypergraph comparison to isolate the effect of event initialization.

B SUPPORT-WINDOW VISUALIZATION

Figure 4 shows the support-window visualization. The validation-selected windows are 3, 7, and 6 for P12, MIMIC-III, and MIMIC-IV, respectively. The selected test means are reported in Table 2. Other test-window evaluations are descriptive and are not used for window selection.

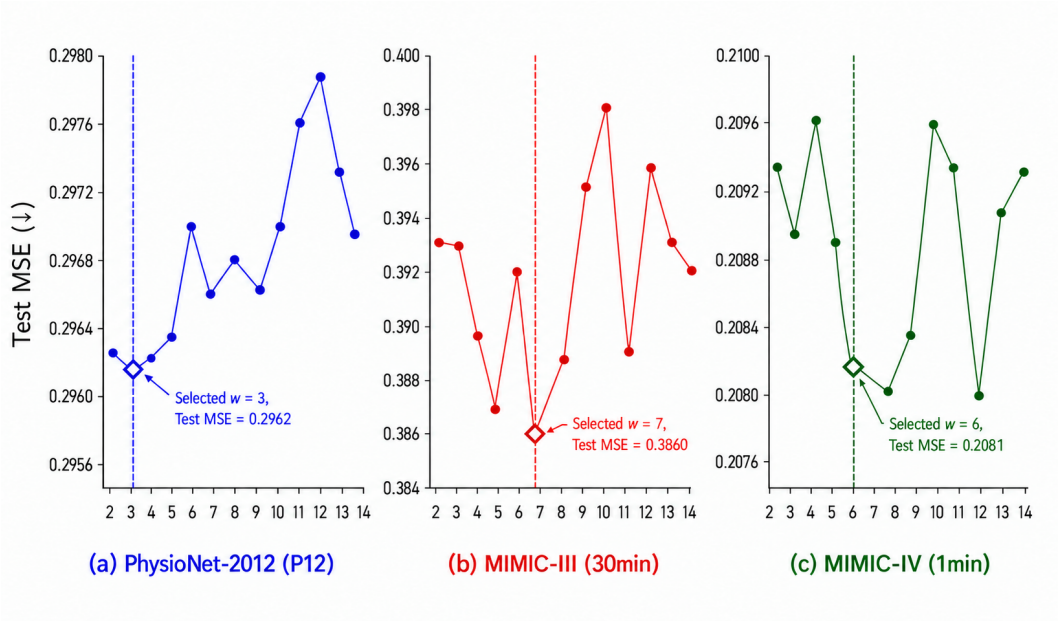

Figure 4: Support-window sensitivity; dashed lines mark validation-selected windows.