DECOMPOSE DYNAMICS BEFORE LEARNING DEPENDENCIES IN SPATIOTEMPORAL SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Relations in networked spatiotemporal systems are often learned from observations that entangle dynamics governed by different mechanisms, obscuring what evolves locally and how it propagates across nodes. We introduce Component-Aware Network Dynamics with Ordered Relations (CANDOR), which decomposes local dynamics before learning their dependencies. CANDOR represents each trajectory through a persistent background, gradual accumulation and release, and sparse shocks. Conditioned on these components, a delay-aware physical branch models edge and sample-dependent propagation over directed topology, while a topology-unconstrained functional branch discovers latent dependencies from background dynamics. Context-adaptive fusion combines functional, forward-propagation, and reverse-support forecasts, with training objectives encouraging specialized and semantically consistent representations. Experiments on two traffic benchmarks and three long-horizon water-quality datasets span two distinct spatiotemporal systems: human-driven urban traffic and naturally evolving river water quality. CANDOR consistently outperforms the strongest baselines, reducing MAE by up to 4.31% for traffic and MSE by up to 5.47% for water-quality forecasting. These results establish decomposition before dependency learning as an effective principle for spatiotemporal representation learning.

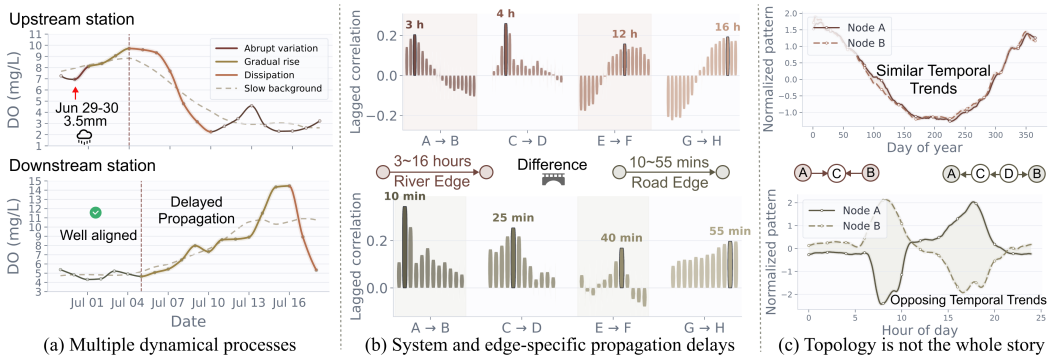
1 INTRODUCTION

Spatiotemporal systems consist of networked entities whose states evolve locally while influencing one another over time. They arise in both human mobility networks and environmental systems, where understanding their dynamics supports applications ranging from traffic management to environmental monitoring (Kumar et al., 2024). The key challenge is to understand how local dynamics evolve and how nodes influence each other, both through physical connections and beyond them.

Recent advances in spatiotemporal forecasting have enabled more flexible modeling of heterogeneous dynamics and adaptive spatial dependencies (Lee & Ko, 2024; Lyu et al., 2025; Gao et al., 2025; Zou et al., 2026). This progress reflects a broader shift from predefined spatial structures and shared temporal models toward learned relationships and context-dependent modeling (Qin et al., 2026). While these developments increase the flexibility of dependency learning, they leave open a central question: how should different local dynamics inform the relationships learned between nodes? As illustrated in Figure 1, local trajectories can combine persistent patterns, gradual changes, and abrupt shocks. Their effects may propagate through physical connections with delays that vary across systems and edges, while nodes without direct connections may also exhibit related temporal patterns. Learning dependencies from a single representation of these mixed dynamics can obscure how individual processes evolve, propagate, and relate across nodes. These observations motivate a central principle: *decompose dynamics before learning dependencies*.

To this end, we introduce **Component-Aware Network Dynamics with Ordered Relations (CANDOR)**. CANDOR decomposes each trajectory into four components: background, accumulation, release, and shock, capturing persistent patterns, gradual buildup, gradual decline, and abrupt changes, respectively. Conditioned on this decomposition, a delay-aware physical branch captures interactions that vary across edges and samples over directed topology, while a functional branch learns latent dependencies beyond direct physical connections. CANDOR then adaptively fuses predictions from the functional, forward propagation, and reverse support branches. Its training objectives

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.



(a) Multiple dynamical processes (b) System and edge-specific propagation delays (c) Topology is not the whole story

Figure 1: Representative characteristics of real-world spatiotemporal systems: (a) coexisting dynamics, (b) edge and system-dependent delays reflected by distinct peak lags, and (c) nonlocal dependencies beyond topology with similar or opposing temporal patterns.

encourage branch specialization and semantic consistency between the decomposed components and their corresponding branch states at each node. Our contributions are threefold:

- **Decomposition before dependency learning.** We introduce a component-aware formulation that disentangles local dynamics into background, accumulation, release, and shock, and uses the resulting components to guide the learning of dependencies between nodes.
- **Mechanism-specific relation learning.** We develop complementary physical and functional branches that capture delay-aware propagation over directed topology and latent dependencies beyond physical connectivity. Adaptive fusion integrates their predictions, while specialization objectives promote distinct and interpretable relational roles.
- **Cross-system validation.** We evaluate CANDOR on urban traffic and river water quality systems to assess its effectiveness across distinct spatiotemporal settings. We also release three water-quality datasets comprising over 1.8 million valid station-hour observations from 43 monitoring sites over five years, together with directed river-network information.

2 Related Works

Dynamics Decomposition for Time Series Modeling. Recent decomposition-based methods separate temporal patterns to facilitate forecasting. TimeMixer (Wang et al., 2024a) and TimeMixer++ (Wang et al., 2025) mix seasonal and trend patterns across scales and resolutions, while TimeKAN models distinct frequency bands with specialized networks (Huang et al., 2025). Leddam combines learnable decomposition with inter-series and intra-series dependency modeling (Yu et al., 2024). Decomposition also guides component-specific masking in ST-MTM (Seo & Lim, 2025) and separate seasonal and trend supervision in DBLoss (Qiu et al., 2026). More recently, CEDAR separates action-conditioned state transitions from event-driven residual corrections (Meng et al., 2026), while a synthetic evaluation suite (Wu et al., 2026) evaluates component recovery under controlled generative mechanisms and connects decomposition quality to downstream structure discovery. These developments extend decomposition from temporal pattern extraction to modeling distinct sources of variation and evaluating component fidelity. CANDOR focuses on how decomposed dynamics inform relationships between nodes. Its background, accumulation, release, and shock components guide physical propagation modeling, while background representations inform the functional relation graph, linking local dynamics decomposition to distinct mechanisms of network dependence.

Graph-Based Dependency and Propagation Modeling. Graph-based forecasting combines structural priors with learned dependencies to capture interactions between nodes (Cini et al., 2025). MegaCRN learns adaptive spatiotemporal graph structures through meta-graph learning (Jiang et al., 2023b), while PDFormer incorporates propagation delays into spatial attention (Jiang et al., 2023a). TESTAM adapts to different traffic conditions through temporal, static-graph, and dynamic-graph experts (Lee & Ko, 2024). Beyond predefined connectivity, DUET learns channel relationships through soft clustering (Qiu et al., 2025), and TimeFilter selects relevant spatiotemporal dependen-

cies at the patch level (Hu et al., 2025). Recent work further expands graph adaptation: MetaDG jointly generates dynamic adjacency matrices and model parameters (Zou et al., 2026), whereas UrbanGraph encodes physical processes into dynamic heterogeneous topology (Xin et al., 2026). Existing methods generally learn a single dependency structure from entangled observations. CANDOR instead assigns distinct relational mechanisms to different dynamical evidence, separating directed, delay-aware physical propagation from topology-free functional dependence. This yields complementary relations with explicit mechanistic roles.

3 METHODOLOGY

3.1 PROBLEM FORMULATION AND OVERVIEW

Let $\mathcal{G}_0 = (\mathcal{V}, \mathcal{E})$ denote a directed physical network with $N = |\mathcal{V}|$ nodes, where $(i, j) \in \mathcal{E}$ represents a connection from node i to node j. Given historical observations $\mathbf{X}_{t-L+1:t} \in \mathbb{R}^{L\times N}$ and $\mathcal{G}_0$, and the ground-truth future $\mathbf{Y}^{\star}_{t+1:t+H} \in \mathbb{R}^{H\times N}$, the task is to predict $\mathbf{Y}_{t+1:t+H} \in \mathbb{R}^{H\times N}$, where t is the current time, L is the history length, and H is the forecasting horizon.

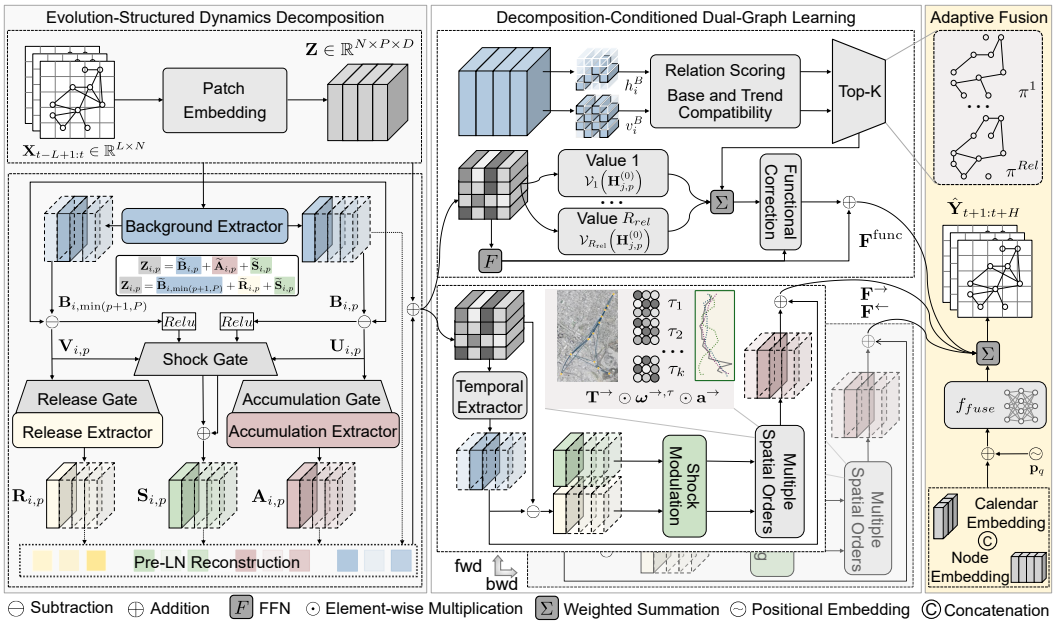


Figure 2: **Overview of CANDOR. (1) Evolution-structured dynamics decomposition** separates persistent and transient dynamics through complementary residual references. **(2) Decomposition-conditioned dual-graph learning** combines delay-aware propagation over directed physical support with background-conditioned functional relations beyond topology. **(3) Context-adaptive fusion** integrates functional, forward, and reverse-support forecasts contextually.

CANDOR decomposes local dynamics before learning dependencies. As shown in Figure 2, it first separates node representations into four components: background, accumulation, release, and shock. These components inform two complementary relational mechanisms: the physical propagation graph captures delayed interactions over directed physical connections, while the functional relation graph uses background representations to infer dependencies beyond physical adjacency. CANDOR then adaptively fuses functional, forward-propagation, and reverse-support representations to generate the forecast. This organization connects local dynamics to complementary and mechanism-specific relations, avoiding dependency learning from entangled observations.

3.2 EVOLUTION-STRUCTURED DYNAMICS DECOMPOSITION

Persistent behavior and local deviations provide different evidence for learning relationships between nodes. We therefore structure local dynamics by both their reference state and their temporal

evolution. A shared background provides complementary references for accumulation and release, while changes in the resulting deviations guide the extraction of abrupt shocks.

We first embed the observed history as $\mathbf{Z} = \mathrm{PatchEmbed}(\mathbf{X}_{t-L+1:t}) \in \mathbb{R}^{N\times P\times D}$, where P is the number of historical patches and D is the latent dimension. A causal temporal convolution $\mathcal{C}_B$ followed by layer normalization LN_B extracts the background, $\mathbf{B} = \mathrm{LN}_B(\mathcal{C}_B(\mathbf{Z}))$. We then construct two residual views to distinguish local buildup from relaxation. The current background measures deviation from the present baseline, whereas the next observed background provides a forward reference for how the state evolves toward the subsequent baseline:

$$\mathbf{U}_{i,p} = \mathbf{Z}_{i,p} - \mathbf{B}_{i,p}, \qquad \mathbf{V}_{i,p} = \mathbf{Z}_{i,p} - \mathbf{B}_{i,\min(p+1,P)}, \tag{1}$$

where i indexes nodes and p indexes historical patches. Both references lie within the observed window, with the final background repeated at the boundary. We use the two residual views to extract accumulation $\mathbf{A}$ and release $\mathbf{R}$, respectively. Figure 3 illustrates the two reference-consistent views and their roles in subsequent propagation. The shared background defines accumulation and release relative to the current and subsequent reference states, while sparse shocks account for abrupt deviations in both views.

First and second-order temporal differences characterize the direction and variation of local changes, guiding adaptive selection within each view under node and calendar context. This combines distinct background references with evolution-dependent selection to provide an inductive bias for local buildup and relaxation. Abrupt disturbances may emerge in either view. Therefore, we extract shock $\mathbf{S}$ from both, using residual and temporal-difference magnitudes to select pronounced deviations and rapid changes regardless of sign. The resulting components provide structured inputs for subsequent relation learning, with their intended roles further reinforced by reconstruction and alignment objectives during training (Section 3.5). Full decomposition computations are provided in Appendix A.1.

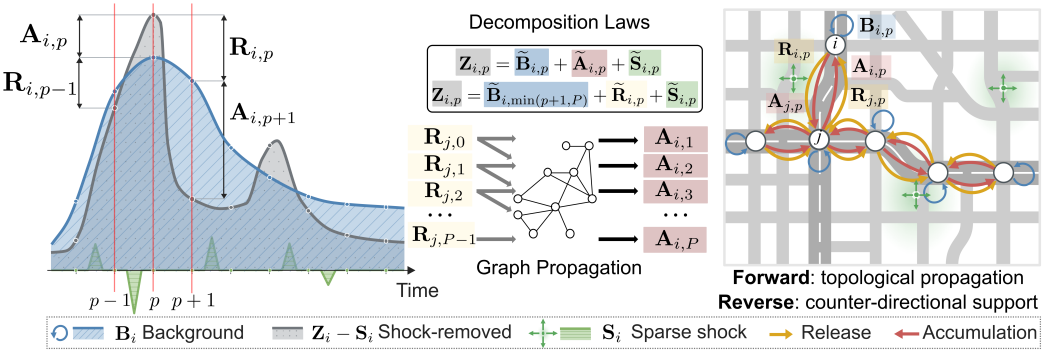


Figure 3: **Evolution-structured dynamics decomposition.** A shared background defines complementary residual views for accumulation and release, while sparse shocks capture abrupt deviations. The components provide distinct local-evolution roles for relation learning. Release-associated residuals propagate over the physical graph and support subsequent downstream accumulation.

3.3 Decomposition-Conditioned Dual-Graph Learning

We integrate the four components through a residual projection to obtain the refined history $\mathbf{H}^{(0)} \in \mathbb{R}^{N\times P\times D}$. The decomposition conditions the two branches in distinct ways: the physical branch propagates states derived from $\mathbf{H}^{(0)}$ over directed topology, whereas the functional branch additionally uses $\mathbf{B}$ to determine relational weights. This gives the components distinct downstream roles: the background additionally serves as a latent signature for inferring non-topological functional relations, while accumulation, release, and shock characterize transient dynamics involved in propagation and state evolution.

Physical propagation graph. Physical connectivity constrains interaction paths but does not specify how neighboring states at different temporal positions contribute. We construct the physical propagation graph $\mathcal{G}_{\mathrm{phys}}$ by combining delay-specific affinities shared across samples with history-conditioned affinities that adapt to each observed window. This separates shared temporal interaction patterns from variations in interaction strength across samples. For forward and reverse support,

indexed by $d \in \{\rightarrow, \leftarrow\}$, first-order aggregation is

$$\mathbf{g}_{i,p}^{d} = \sum_{\tau=1}^{K_t} \sum_{j \neq i} W_{ij}^{d,\tau} \mathbf{\Psi}_{j,p,\tau}^{d}, \qquad \mathbf{W}^{d,\tau} = \mathrm{RowNorm}\big(\mathbf{T}^{d} \odot \boldsymbol{\omega}^{d,\tau} \odot \mathbf{a}^{d}\big), \tag{2}$$

where K_t is the temporal window size and $\mathbf{\Psi}_{j,p,\tau}^{d}$ is the transformed propagation feature at position τ within the window ending at patch p, derived from $\mathbf{H}^{(0)}$. The directional support $\mathbf{T}^{d}$ is obtained from $\mathcal{G}_0$. The affinities $\boldsymbol{\omega}^{d,\tau}$ are learned from directional node embeddings and vary across edges and temporal positions while remaining shared across samples. In contrast, $\mathbf{a}^{d}$ is computed from historical observations and calendar context for each sample and is shared across temporal positions. $\odot$ denotes element-wise multiplication, and $\mathrm{RowNorm}$ normalizes over source nodes while preserving zero rows. In this case, $\boldsymbol{\omega}^{d,\tau}$ and $\mathbf{a}^{d}$ capture temporal-position structure shared across samples and sample-specific variation across observed windows, respectively.

We extend aggregation to multiple spatial orders to capture interactions along longer paths. Forward follows directed physical connections, while reverse supplies predictive context from downstream states. The two paths yield forecast representations $\mathbf{F}^{\rightarrow}, \mathbf{F}^{\leftarrow} \in \mathbb{R}^{N \times P_{\mathrm{out}} \times D_{\mathrm{f}}}$, where P_{out} is the number of target patches and D_{f} is the forecast representation dimension.

Functional relation graph. Functional dependencies can extend beyond physical topology and arise from shared background dynamics (e.g., similar persistent temporal patterns among non-adjacent nodes). In this case, we construct the functional relation graph $\mathcal{G}_{\mathrm{func}}$ from background dynamics, evaluating compatibility in both background state and evolution. For node i, a descriptor $\mathbf{h}_i^{B}$ summarizes background level and trajectory shape with node and calendar context; a separate descriptor $\mathbf{v}_i^{B}$ summarizes background trends. Multiple relation channels capture different compatibility patterns in these descriptors. For each channel r, we compute

$$s_{ij}^{r} = (\mathbf{h}_i^{B})^{\top} \mathbf{M}_r \mathbf{h}_j^{B} + \eta (\mathbf{v}_i^{B})^{\top} \mathbf{N}_r \mathbf{v}_j^{B} + b_r, \qquad \pi_{ij}^{r} = \mathrm{TopKSoftmax}_{(j,r)}(s_{ij}^{r}), \tag{3}$$

where $\mathbf{M}_r$ and $\mathbf{N}_r$ are learned relation-specific matrices, b_r is a bias, and η weights trend compatibility. For each receiving node, $\mathrm{TopKSoftmax}$ jointly selects the top k neighbor-relation pairs and normalizes their scores without a physical adjacency mask. This jointly determines which nodes contribute and through which relation channels.

The weights π_{ij}^{r} aggregate relation-specific transformations of $\mathbf{H}^{(0)}$ to refine the temporal forecast, yielding $\mathbf{F}^{\mathrm{func}} \in \mathbb{R}^{N \times P_{\mathrm{out}} \times D_{\mathrm{f}}}$. Background dynamics guide relation selection, while all four components contribute to information exchange. The three forecast representations ($\mathbf{F}^{\mathrm{func}}$, $\mathbf{F}^{\rightarrow}$, and $\mathbf{F}^{\leftarrow}$) are then passed to adaptive fusion. Full computations are provided in Appendix A.2.

3.4 Context-Adaptive Forecast Fusion

The relative contributions of functional dependence, forward propagation, and reverse-support context can vary across nodes and forecasting positions. Therefore, we condition their fusion on node identity, target calendar information, and relative forecast position, allowing the three mechanisms to contribute differently across prediction contexts.

Let q index target patches and $\mathcal{B} = \{\mathrm{func}, \rightarrow, \leftarrow\}$ denote the three branches. We construct a context vector $\mathbf{c}_{i,q}$ by concatenating a learned node embedding, target time-of-day and day-of-week embeddings, and a sinusoidal encoding of q. An MLP f_{fuse} maps this context to three logits, which determine the fused representation:

$$\boldsymbol{\lambda}_{i,q} = \mathrm{softmax}(f_{\mathrm{fuse}}(\mathbf{c}_{i,q})), \qquad \mathbf{F}_{i,q}^{\mathrm{fuse}} = \sum_{b \in \mathcal{B}} \lambda_{i,q}^{b} \mathbf{F}_{i,q}^{b}, \tag{4}$$

where $\lambda_{i,q}^{b}$ is the weight of branch b and it is shared across its latent features. Calendar context captures recurring temporal variation, while positional encoding distinguishes forecasting positions even when their calendar contexts coincide. A shared prediction head decodes $\mathbf{F}^{\mathrm{fuse}}$ into patch-level predictions, which are averaged over overlaps to obtain the final forecast $\hat{\mathbf{Y}}_{t+1:t+H}$. Further details on adaptive branch fusion and patch-level forecast decoding are provided in Appendix A.3.

3.5 LEARNING OBJECTIVES

Prediction supervision alone does not ensure that decomposed components retain their intended roles during relation learning. We constrain both how they explain local dynamics and how they align with physical propagation states.

Let $\widetilde{\mathbf{B}}$, $\widetilde{\mathbf{A}}$, $\widetilde{\mathbf{R}}$, and $\widetilde{\mathbf{S}}$ denote component outputs before normalization. We impose reconstruction constraints consistent with the two background references:

$$\begin{aligned}\mathcal{L}_{\mathrm{rec}} = \frac{1}{2}\,\mathrm{SL1}\Big(\widetilde{\mathbf{B}} + \widetilde{\mathbf{A}} + \widetilde{\mathbf{S}}, \mathrm{sg}(\mathbf{Z})\Big) \\ + \frac{1}{2}\,\mathrm{SL1}\Big(\widetilde{\mathbf{B}}_{:,2:P} + \widetilde{\mathbf{R}}_{:,1:P-1} + \widetilde{\mathbf{S}}_{:,1:P-1}, \mathrm{sg}(\mathbf{Z}_{:,1:P-1})\Big)\,,\end{aligned} \tag{5}$$

where $\mathrm{SL1}$ is mean smooth-L_1 loss and sg stops gradients through its argument. Accumulation explains a local state relative to its current background, whereas release explains it relative to the next background, consistent with the two residual references in Eq. 1. Shock contributes to both reconstructions. These soft constraints give the two residual views distinct learning targets.

We further introduce a role-consistency loss $\mathcal{L}_{\mathrm{role}}$ to connect the decomposition with physical-path states. At each node, projected cosine similarity aligns background at the next patch with the current temporal states, and accumulation at the next patch with incoming graph contributions $\mathbf{o}^{\Sigma}$. Release is aligned with the corresponding propagation residuals $\mathbf{E}^{\Sigma}$ at the same patch. Forward and reverse states are combined for these comparisons. This encourages the components to retain their intended roles in local prediction and relational propagation. The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{pred}} + \mathcal{L}_{\mathrm{aux}} + \chi(e)\,(\lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \mathcal{L}_{\mathrm{role}})\,, \tag{6}$$

where $\mathcal{L}_{\mathrm{pred}}$ supervises the final forecast. The weighted auxiliary objective $\mathcal{L}_{\mathrm{aux}}$ supervises each branch before fusion, enforces historical prediction consistency, and discourages redundancy between temporal and propagation states. It also encourages smooth background evolution and selective shock activation. The coefficient λ_{rec} weights reconstruction, and $\chi(e)$ activates reconstruction and role consistency at the dataset-specific epoch e_0. Full definitions are provided in Appendix A.4.

4 EXPERIMENTS

Tasks and datasets. We evaluate CANDOR on two contrasting spatiotemporal systems: human-driven urban traffic and natural river networks governed by directed hydrologic transport. Traffic combines recurrent mobility patterns with abrupt congestion, whereas river systems exhibit sparse directional topology and delayed downstream effects, providing distinct tests of local dynamics and cross-node propagation. For traffic forecasting, we use METR-LA (Jagadish et al., 2014) and PEMS-BAY (Chen et al., 2001), containing 207 and 325 sensors, respectively, and predict 12 five-minute steps from 12 historical steps. We further introduce three five-year datasets of hourly dissolved-oxygen observations from BASIN030501 (Santee), BASIN031300 (Apalachicola), and BASIN180102 (Klamath). They contain 1.88 million station-hour measurements from 43 monitoring sites and 87 directed river connections. We use 48 historical hours to forecast the following 240 hours. Chronological train/validation/test splits are 70/10/20 for traffic and 60/20/20 for water quality. Dataset construction and quality-control procedures are provided in Appendix A.5.

Implementation details. CANDOR uses a hidden dimension of 32, a forecast representation dimension of 256, and a dropout rate of 0.1, resulting in fewer than 0.91M parameters on every dataset. The physical branch uses a two-patch propagation window and two spatial orders to capture delayed and multi-hop propagation with limited overhead. The functional branch retains the top 16 node-relation pairs for each receiving node from up to four relation channels, encouraging sparse dependency learning. Patch length and stride are $4/2$ for METR-LA and $3/3$ for PEMS-BAY; for water quality, they scale with the forecast horizon and reach $16/8$ from 192 hours onward. We train CANDOR with Adam at a learning rate of 2×10^{-3}. Complete hyperparameter settings, sensitivity analyses, and time-complexity analysis are provided in Appendix A.6.

Baselines and evaluation protocol. We compare CANDOR with 15 representative baselines from two complementary model families. This choice reflects the heterogeneity of spatiotemporal systems: graph-based models are advantageous when cross-node propagation is informative, whereas

general time-series models can be more effective when spatial coupling is weak or noisy and node-wise temporal regularities dominate. Spatiotemporal forecasting methods include TESTAM (Lee & Ko, 2024), AutoSTF (Lyu et al., 2025), ST-SSDL (Gao et al., 2025), HyperD (Shao et al., 2026), RSTIB-MLP (Chen et al., 2025), STPGNN (Kong et al., 2024), MSTHG (Zhang et al., 2026), and FaST (Zhao et al., 2026). They cover adaptive graph learning, propagation-aware modeling, hypergraph dependencies, and multiscale spatiotemporal representations. General time-series forecasters include DTAF (Lu et al., 2026), LatentTSF (Yang et al., 2026), ModernTCN (Luo & Wang, 2024), Pathformer (Chen et al., 2024), PhaseFormer (Niu et al., 2026), TimeBridge (Liu et al., 2025), and TimeMixer++ (Wang et al., 2025). These methods span convolutional, decomposition-based, frequency-aware, and multiscale architectures. Including both families enables evaluation across systems with different degrees of propagation strength without assuming that spatial modeling is uniformly beneficial. Detailed descriptions for all baselines are provided in Appendix A.7.

All methods use identical chronological splits, history lengths, forecast horizons, targets, and evaluation pipelines. Each experiment is repeated using the same three seeds. Traffic forecasting uses MAE as the primary metric, with RMSE and MAPE as complementary measures; water-quality forecasting uses MSE as the primary metric and MAE as a complementary measure. All metrics are computed on the original data scale. For readability, the main tables report the top eight baselines: traffic models are selected separately by unrounded overall MAE, while a shared water-quality set is ranked by MSE averaged across all basins and horizons. Complete results for all 15 baselines and all water-quality horizons, including mean$\pm$std over three seeds, are provided in Appendix A.8.

4.1 Main Results

Table 1: Mean forecasting performance over three seeds on METR-LA and PEMS-BAY. For each dataset, the eight strongest baselines are independently selected from 15 candidates according to mean MAE. Best results are in **bold** and second-best results are underlined.

METR-LA

Method	Overall			15 min			30 min			60 min		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
TESTAM (Lee & Ko, 2024)	2.982	6.136	8.14%	2.670	5.206	6.89%	3.017	6.189	8.25%	3.420	7.215	9.86%
ST-SSDL (Gao et al., 2025)	2.986	6.191	8.13%	2.643	5.133	6.80%	3.021	6.225	8.23%	3.474	7.429	10.00%
AutoSTF (Lyu et al., 2025)	2.999	6.195	8.16%	2.692	5.231	6.97%	3.040	6.254	8.29%	3.427	7.292	9.81%
MSTHG (Zhang et al., 2026)	3.129	6.357	8.84%	2.790	5.388	7.42%	3.177	6.445	9.03%	3.602	7.460	10.71%
HyperD (Shao et al., 2026)	3.370	6.721	10.29%	3.091	5.956	9.20%	3.451	6.832	10.57%	3.778	7.747	12.04%
STPGNN (Kong et al., 2024)	3.485	6.888	10.12%	2.977	5.706	8.12%	3.499	6.911	10.18%	4.257	8.339	13.08%
RSTIB-MLP (Chen et al., 2025)	3.778	9.082	10.12%	3.221	7.434	8.36%	3.856	9.265	10.39%	4.522	10.836	12.41%
FaST (Zhao et al., 2026)	3.911	8.932	10.38%	3.191	6.977	8.26%	3.912	8.850	10.45%	4.976	11.231	13.40%
CANDOR (Ours)	**2.854**	**5.860**	**7.77%**	**2.548**	**4.892**	**6.47%**	**2.889**	**5.891**	**7.86%**	**3.300**	**6.991**	**9.62%**

PEMS-BAY

Method	Overall			15 min			30 min			60 min		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
TESTAM (Lee & Ko, 2024)	1.530	3.566	3.44%	1.297	2.781	2.74%	1.593	3.681	3.59%	1.849	4.320	4.34%
AutoSTF (Lyu et al., 2025)	1.548	3.599	3.44%	1.298	2.758	2.72%	1.613	3.712	3.61%	1.881	4.389	4.36%
ST-SSDL (Gao et al., 2025)	1.579	3.678	3.51%	1.304	2.765	2.70%	1.642	3.762	3.65%	1.946	4.555	4.55%
MSTHG (Zhang et al., 2026)	1.586	3.683	3.59%	1.323	2.816	2.82%	1.651	3.801	3.75%	1.935	4.490	4.56%
RSTIB-MLP (Chen et al., 2025)	1.600	3.716	3.61%	1.348	2.870	2.84%	1.665	3.832	3.77%	1.940	4.532	4.60%
FaST (Zhao et al., 2026)	1.720	4.153	3.81%	1.366	2.983	2.86%	1.762	4.147	3.92%	2.232	5.368	5.13%
HyperD (Shao et al., 2026)	1.796	4.275	4.45%	1.643	3.889	4.07%	1.856	4.363	4.62%	2.109	4.948	5.24%
TimeBridge (Liu et al., 2025)	1.911	4.581	4.26%	1.425	3.095	2.97%	1.922	4.475	4.26%	2.631	6.136	6.15%
CANDOR (Ours)	**1.508**	**3.500**	**3.34%**	**1.257**	**2.647**	**2.59%**	**1.565**	**3.582**	**3.46%**	**1.846**	**4.308**	**4.31%**

Traffic forecasting. Table 1 shows that CANDOR achieves the best mean performance across both traffic datasets and all reported forecasting horizons. On METR-LA, it reduces Overall MAE from 2.982 to 2.854 (4.31%) and Overall RMSE from 6.136 to 5.860 (4.49%) relative to the strongest baseline. The improvement remains 3.51% in MAE at the 60-min horizon, demonstrating sustained accuracy under longer-range propagation. On PEMS-BAY, where competing methods are more tightly clustered, CANDOR still reduces overall MAE and RMSE by 1.46% and 1.86%, respectively, while ranking first at every horizon. These results show that CANDOR benefits both heterogeneous congestion dynamics and smoother traffic regimes.

Table 2: Mean long-horizon forecasting performance over three seeds. The same eight baselines are selected from 15 candidates according to their MSE averaged across all three basins and all reported forecasting horizons. Complete results for all methods and horizons are provided in Appendix A.8.

Model		FaST (2026)		MSTHG (2026)		TimeMixer++ (2025)		LatentTSF (2026)		Pathformer (2024)		PhaseFormer (2026)		DTAF (2026)		TimeBridge (2025)		CANDOR Ours	
Year																			
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
030501	48	0.308	0.336	0.347	0.359	0.313	0.342	0.321	0.357	0.311	0.334	0.317	0.344	0.320	0.344	0.311	0.339	**0.289**	**0.328**
	144	0.604	0.495	0.558	0.494	0.611	0.502	0.613	0.523	0.608	0.495	0.619	0.507	0.623	0.505	0.618	0.502	**0.544**	**0.478**
	240	0.802	0.589	0.706	0.570	0.811	0.598	0.799	0.616	0.807	0.589	0.815	0.599	0.821	0.598	0.820	0.596	**0.665**	**0.549**
031300	48	0.213	0.294	0.227	0.310	0.217	0.299	0.248	0.332	0.220	0.298	0.231	0.312	0.237	0.314	0.222	0.301	**0.205**	**0.287**
	144	0.441	0.454	0.469	0.481	0.447	0.460	0.468	0.482	0.451	0.461	0.465	0.470	0.471	0.472	0.468	0.468	**0.420**	**0.449**
	240	0.557	0.525	0.585	0.552	0.559	0.528	0.578	0.548	0.581	0.537	0.584	0.539	0.591	0.542	0.593	0.539	**0.528**	**0.520**
180102	48	0.725	0.441	0.748	0.460	0.726	0.443	0.739	0.473	0.732	0.440	0.752	0.455	0.743	0.451	0.745	0.445	**0.712**	**0.440**
	144	1.390	0.652	1.404	0.688	1.394	0.659	1.394	0.701	1.409	0.653	1.440	0.676	1.429	0.672	1.418	0.660	**1.302**	**0.644**
	240	1.879	0.789	1.912	0.835	1.901	0.800	1.846	0.835	1.890	0.791	1.931	0.814	1.924	0.810	1.950	0.803	**1.763**	**0.780**

Water-quality forecasting. Table 2 presents a more challenging setting with sparse directed networks. CANDOR achieves the lowest MSE and MAE at every reported horizon on all three basins. Its horizon-averaged MSE improvements over the strongest baselines are 5.47%, 4.68%, and 5.29% on the three datasets, respectively. Unlike traffic forecasting, generic spatiotemporal models do not consistently dominate temporal forecasters on these datasets. This suggests that sparse topology and strong node-specific temporal regularities can limit uniform spatial aggregation. CANDOR addresses this regime by decomposing local dynamics before dependency learning, modeling delayed interactions over directed physical support, and complementing them with functional relations beyond topology. The resulting advantage across basins and horizons shows that CANDOR can exploit spatial propagation without assuming that it is uniformly informative. The improvements are robust across the three seeds (Appendix A.8): CANDOR's worst-seed result outperforms the best seed of every baseline on both traffic datasets and in 222 of 225 water-quality comparisons.

4.2 Ablation Studies

Figure 4 evaluates the modeling chain on two representative datasets; results on all five datasets are reported in Appendix A.9. The "w/o Decomp." variant removes the decomposition and passes patch representations directly to both relational branches, whereas "w/o Aux. loss" retains the components but removes their reconstruction and role-consistency objectives. "w/o Physical" and "w/o Functional" remove the two relational branches, while "Forward only" and "Reverse only" retain one physical direction. "Raw adjacency" replaces the learned physical operator with normalized topology. Finally, "w/o Relation" removes functional relation modeling, while "avg Fusion" replaces adaptive fusion with uniform averaging.

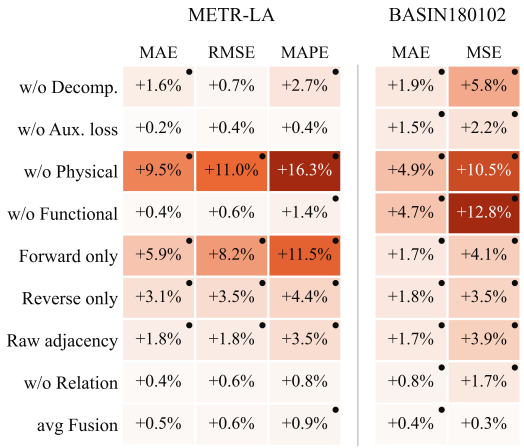
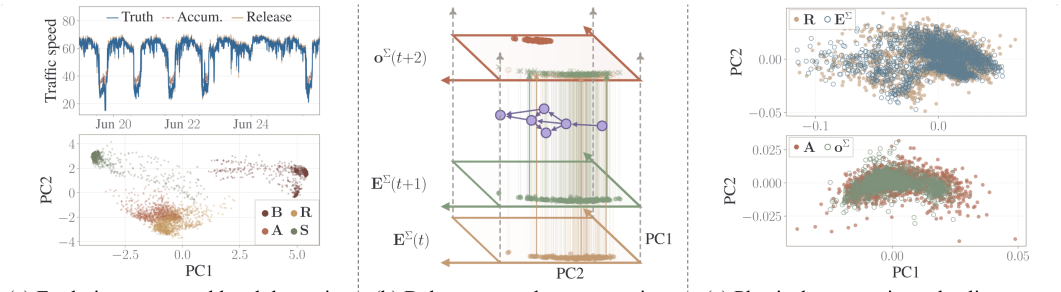


Figure 4: Seed-paired relative degradation (%) under each ablation variant. Dots denote changes exceeding twice the full-model seed variation.

The results support the progression from decomposition to relational learning. Removing decomposition increases METR-LA MAE by 1.6% and BASIN180102 MSE by 5.8%; removing only its auxiliary objectives increases BASIN180102 MAE/MSE by $1.5\%/2.2\%$. This shows that the benefit arises from both component representations and their assigned roles. The relational branches are complementary across systems. Removing physical propagation increases METR-LA MAE/MAPE by $9.5\%/16.3\%$, whereas removing the physical and functional branches on BASIN180102 increases MSE by 10.5% and 12.8%, respectively. Both single-direction variants degrade performance. Replacing the learned operator with raw adjacency further increases METR-LA MAPE by 3.5% and BASIN180102 MSE by 3.9%, confirming that topology alone does not capture direction, delay, and history-dependent propagation. Relation modeling and adaptive fusion contribute

up to 3.4% and 2.4% in MSE in the complete matrix. Together with the role alignments in Figure 5, these results support CANDOR's central premise: it organizes local dynamics into evolution-specific roles, enabling ordered physical propagation and complementary functional relations to be learned and adaptively combined.

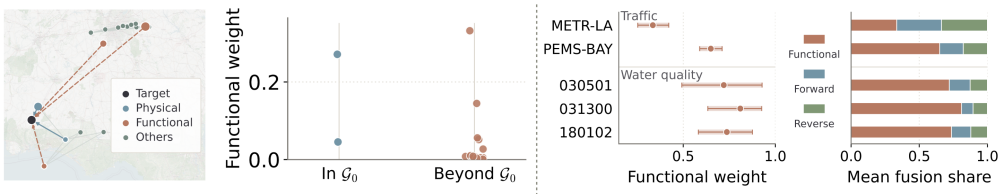
4.3 Model Analysis



(a) Evolution-structured local dynamics (b) Delay-aware edge propagation (c) Physical propagation role alignment

Figure 5: Decomposition and physical propagation on METR-LA. (a) Reference-consistent reconstruction and embeddings of the four components. (b) Delayed mapping from release-associated residuals $\mathbf{E}^{\Sigma}$ to incoming graph contribution $\mathbf{o}^{\Sigma}$. (c) Role alignment of $\mathbf{R}$ with $\mathbf{E}^{\Sigma}$ and $\mathbf{A}$ with $\mathbf{o}^{\Sigma}$.

Figure 5 validates the bridge from decomposition to physical propagation. Panel (a) shows that the reference-consistent reconstructions preserve the trajectory while background, accumulation, release, and shock form distinct embedding regions, demonstrating information retention and role differentiation. Let $\mathbf{E}^{\Sigma}$ and $\mathbf{o}^{\Sigma}$ denote the combined propagation residual and incoming graph contribution. Panel (b) maps $\mathbf{E}^{\Sigma}(t)$ and $\mathbf{E}^{\Sigma}(t+1)$ to the later $\mathbf{o}^{\Sigma}(t+2)$, while panel (c) aligns $\mathbf{R}$ with $\mathbf{E}^{\Sigma}$ and $\mathbf{A}$ with $\mathbf{o}^{\Sigma}$. These results support delayed propagation from release to subsequent accumulation. Figure 6 examines their complementarity with physical propagation. Panel (a) shows that background-conditioned functional relations extend beyond $\mathcal{G}_0$, capturing persistent nonlocal dependencies without transient deviations. Panel (b) shows context-dependent allocation: METR-LA favors physical propagation, whereas PEMS-BAY and the water-quality datasets assign more weight to functional relations. This indicates that fusion preserves distinct pathways and supplements topology when needed. Further validation and physical interpretations are provided in Appendix A.10.



(a) Background-defined nonlocal relations (b) Context-adaptive relation fusion

Figure 6: Functional relations beyond topology and context-adaptive fusion. (a) Background dynamics identify nonlocal functional relations beyond physical adjacency. (b) CANDOR adjusts the use of functional, forward, and reverse-support forecasts across systems.

5 Conclusion

We introduce **CANDOR**, a component-aware framework that decomposes local dynamics before learning node dependencies. By separating persistent, evolving, and abrupt dynamics, CANDOR distinguishes local evolution from cross-node interaction and combines delayed physical propagation with nonlocal functional dependence. Across human-driven traffic and natural river systems, CANDOR reduces traffic MAE by up to 4.31% and water-quality MSE by up to 5.47%. Water-quality results further show that spatiotemporal models do not consistently outperform temporal forecasters under sparse topology and heterogeneous delays. These results support decomposition before dependency learning as a general principle for spatiotemporal representation learning.

AI USE STATEMENT

In this work, we used generative AI tools for language editing, including improving grammar, phrasing, and readability of the manuscript. All AI-assisted text was reviewed and revised by the authors to ensure that it accurately reflects the methods, experiments, and claims in this paper. The authors take responsibility for the final content of this work, including all text, claims, data, and artifacts.

ETHICS STATEMENT

This work does not involve human participants or personally identifiable information. The proposed method is intended to support domain experts in applications such as traffic management and environmental monitoring. Forecasting errors or incomplete observations may affect downstream decisions; users should therefore consider data quality, uncertainty, and local operational constraints. We will document the released water-quality datasets and their processing procedures to support responsible reuse.

REPRODUCIBILITY STATEMENT

The main paper specifies the problem formulation, model design, and experimental protocol, while the Appendix provides complete implementation details, training objectives, and additional analyses. The supplementary material includes the implementation and configuration files.

REFERENCES

Chao Chen, Karl Petty, Alexander Skabardonis, Pravin Varaiya, and Zhanfeng Jia. Freeway performance measurement system: mining loop detector data. *Transportation research record*, 1748 (1):96–102, 2001.

Min Chen, Guansong Pang, Wenjun Wang, and Cheng Yan. Information bottleneck-guided MLPs for robust spatial-temporal forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 8821–8855. PMLR, 2025. URL https://proceedings.mlr.press/v267/chen25ax.html.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=lJkOCMP2aW.

Xinyang Chen, Huidong Jin, Yu Huang, and Zaiwen Feng. XLinear: A lightweight and accurate mlp-based model for long-term time series forecasting with exogenous inputs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24):20325–20335, 2026. doi: 10.1609/aaai.v40i24.39121. URL https://ojs.aaai.org/index.php/AAAI/article/view/39121.

Andrea Cini, Ivan Marisca, Daniele Zambon, and Cesare Alippi. Graph deep learning for time series forecasting. *ACM Computing Surveys*, 57(12):1–34, 2025.

Haotian Gao, Zheng Dong, Jiawei Yong, Shintaro Fukushima, Kenjiro Taura, and Renhe Jiang. How different from the past? spatio-temporal time series forecasting with self-supervised deviation learning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/d7b351608d824a4680344a02b180a947-Abstract-Conference.html.

Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. Timefilter: Patch-specific spatial-temporal graph filtration for time series forecasting. *arXiv preprint arXiv:2501.13041*, 2025.

Songtao Huang, Zhen Zhao, Can Li, and Lei Bai. Timekan: Kan-based frequency decomposition learning architecture for long-term time series forecasting. In *International Conference on Learning Representations*, volume 2025, pp. 28514–28529, 2025.

Hosagrahar V Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Communications of the ACM*, 57(7):86–94, 2014.

Jiawei Jiang, Chengkai Han, Wayne Xin Zhao, and Jingyuan Wang. Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 4365–4373, 2023a.

Renhe Jiang, Zhaonan Wang, Jiawei Yong, Puneet Jeph, Quanjun Chen, Yasumasa Kobayashi, Xuan Song, Shintaro Fukushima, and Toyotaro Suzumura. Spatio-temporal meta-graph learning for traffic forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 8078–8086, 2023b.

Weiyang Kong, Ziyu Guo, and Yubao Liu. Spatio-temporal pivotal graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8627–8635, 2024. doi: 10.1609/aaai.v38i8.28707. URL https://ojs.aaai.org/index.php/AAAI/article/view/28707.

Rahul Kumar, Manish Bhanu, João Mendes-Moreira, and Joydeep Chandra. Spatio-temporal predictive modeling techniques for different domains: a survey. *ACM Computing Surveys*, 57(2): 1–42, 2024.

Hyunwook Lee and Sungahn Ko. TESTAM: A time-enhanced spatio-temporal attention model with mixture of experts. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=N0nTk5BSvO.

Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. TimeBridge: Non-stationarity matters for long-term time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 39815–39840. PMLR, 2025. URL https://proceedings.mlr.press/v267/liu25cb.html.

Junkai Lu, Peng Chen, Chenjuan Guo, Yang Shu, Meng Wang, and Bin Yang. Towards non-stationary time series forecasting with temporal stabilization and frequency differencing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24070–24078, 2026. doi: 10.1609/aaai.v40i29.39585. URL https://ojs.aaai.org/index.php/AAAI/article/view/39585.

Donghao Luo and Xue Wang. ModernTCN: A modern pure convolution structure for general time series analysis. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=vpJMJerXHU.

Tengfei Lyu, Weijia Zhang, Jinliang Deng, and Hao Liu. AutoSTF: Decoupled neural architecture search for cost-effective automated spatio-temporal forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025. doi: 10.1145/3690624.3709323. URL https://doi.org/10.1145/3690624.3709323.

Junjie Meng, Ranxu Zhang, Zi-an Zhang, Shujun Liu, Xiaoning Qi, Xiaozhou Xu, Yanyong Zhang, Hui Xiong, and Chao Wang. Cedar: Controlled and event-driven demand forecasting via residual decomposition. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 7797–7808, 2026.

Yiming Niu, Jinliang Deng, and Yongxin Tong. PhaseFormer: From patches to phases for efficient and effective time series forecasting. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Lk9SqMQzhX.

Yanjun Qin, Yuchen Fang, Xinke Jiang, Hao Miao, and Xiaoming Tao. Adaptive frequency pathways for spatiotemporal forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15653–15661, 2026.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD conference on knowledge discovery and data mining v. 1*, pp. 1185–1196, 2025.

Xiangfei Qiu, Xingjian Wu, Hanyin Cheng, Xvyuan Liu, Chenjuan Guo, Jilin Hu, and Bin Yang. Dbloss: Decomposition-based loss function for time series forecasting. *Advances in Neural Information Processing Systems*, 38:27741–27768, 2026.

Hyunwoo Seo and Chiehyeon Lim. St-mtm: Masked time series modeling with seasonal-trend decomposition for time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 1209–1220, 2025.

Minlan Shao, Zijian Zhang, Yili Wang, Yiwei Dai, Xu Shen, and Xin Wang. HyperD: Hybrid periodicity decoupling framework for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15689–15697, 2026. doi: 10.1609/aaai.v40i18.38599. URL https://ojs.aaai.org/index.php/AAAI/article/view/38599.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 38626–38652, 2024a.

Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. Timemixer++: A general time series pattern machine for universal predictive analysis. In *International Conference on Learning Representations*, volume 2025, pp. 16980–17016, 2025.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. TimeXer: Empowering transformers for time series forecasting with exogenous variables. In *Advances in Neural Information Processing Systems*, volume 37, 2024b. doi: 10.52202/079017-0015. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0113ef4642264adc2e6924a3cbbdf532-Abstract-Conference.html.

Zipeng Wu, Jiani Wei, Shiqiao Zhou, Fabian Spill, JW Andrews, et al. Time-series decomposition as a standalone task: A mechanism-driven diagnostic benchmark. In *Forty-third International Conference on Machine Learning*, 2026.

Weilin Xin, Chenyu Huang, Peilin Li, Jing Zhong, and Jiawei Yao. Urbangraph: Physics-informed spatio-temporal dynamic heterogeneous graphs for urban microclimate prediction. In *International Conference on Learning Representations*, volume 2026, pp. 58856–58889, 2026.

Jie Yang, Yifan Hu, Yuante Li, Kexin Zhang, Kaize Ding, and Philip S. Yu. From observations to states: Latent time series forecasting. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=s49fw3BVU0.

Guoqi Yu, Jing Zou, Xiaowei Hu, Angelica I Aviles-Rivero, Jing Qin, and Shujun Wang. Revitalizing multivariate time series forecasting: Learnable decomposition with inter-series dependencies and intra-series variations modeling. *arXiv preprint arXiv:2402.12694*, 2024.

Ping Zhang, Jiayu Leng, Liang Yang, Anchen Li, Xiaochun Cao, and Riting Xia. Multi-source information driven spatio-temporal hypergraph learning for traffic forecasting. In *Proceedings of the ACM Web Conference 2026*, pp. 580–591. Association for Computing Machinery, 2026. doi: 10.1145/3774904.3792119. URL https://doi.org/10.1145/3774904.3792119.

Yiji Zhao, Zihao Zhong, Ao Wang, Haomin Wen, Ming Jin, Yuxuan Liang, Huaiyu Wan, and Hao Wu. FaST: Efficient and effective long-horizon forecasting for large-scale spatial-temporal graphs via mixture-of-experts. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1975–1986. Association for Computing Machinery, 2026. doi: 10.1145/3770854.3780165. URL https://doi.org/10.1145/3770854.3780165.

Yiqing Zou, Hanning Yuan, Qianyu Yang, Ziqiang Yuan, Shuliang Wang, and Sijie Ruan. Meta dynamic graph for traffic flow prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 16584–16592, 2026. doi: 10.1609/aaai.v40i19.38699. URL https://ojs.aaai.org/index.php/AAAI/article/view/38699.

A Appendix

A.1 Evolution-Structured Dynamics Decomposition

We detail the computations in Section 3.2, omitting the batch dimension.

Background and residual construction. We divide each node's history into patches of length ℓ and stride s, giving $P = \lfloor (L-\ell)/s \rfloor + 1$. For the observation vector $\mathbf{x}_{i,p} \in \mathbb{R}^{\ell}$ of node i at patch p, the embedding and background are

$$\mathbf{Z}_{i,p} = \mathrm{LN}_Z(\mathbf{W}_Z \mathbf{x}_{i,p} + \mathbf{b}_Z), \qquad \widetilde{\mathbf{B}} = \mathcal{C}_B(\mathbf{Z}), \qquad \mathbf{B} = \mathrm{LN}_B(\widetilde{\mathbf{B}}), \tag{7}$$

where $\mathbf{W}_Z \in \mathbb{R}^{D \times \ell}$ and $\mathbf{b}_Z \in \mathbb{R}^D$ are learned parameters. The operator $\mathcal{C}_B$ is a depthwise temporal convolution with kernel size K and $K-1$ zeros padded on the left.

Equation 1 constructs $\mathbf{U}$ and $\mathbf{V}$ from $\mathbf{B}$. The shifted reference repeats the final background at the boundary and accesses only observed history. Thus, although the convolution is causal, the complete decomposition uses context across the observed window.

Evolution-dependent component extraction. For either residual view $\mathbf{Q} \in \{\mathbf{U}, \mathbf{V}\}$, temporal differences are defined recursively:

$$(\Delta \mathbf{Q})_{i,p} = \begin{cases} \mathbf{0}, & p = 1, \\ \mathbf{Q}_{i,p} - \mathbf{Q}_{i,p-1}, & p > 1, \end{cases} \qquad \Delta^2 \mathbf{Q} = \Delta(\Delta \mathbf{Q}). \tag{8}$$

Let $\mathbf{e}_i \in \mathbb{R}^D$ be a learned node embedding and $\mathbf{m}_p \in \mathbb{R}^D$ the calendar context at the patch start. The latter is obtained by an MLP over concatenated time-of-day and day-of-week embeddings, using 288 and seven categories, respectively. With an affine node projection $\mathcal{P}_E$, the gate input is

$$\boldsymbol{\xi}_{i,p}(\mathbf{Q}) = \left[\mathbf{Q}_{i,p} \| (\Delta \mathbf{Q})_{i,p} \| - (\Delta \mathbf{Q})_{i,p} \| (\Delta^2 \mathbf{Q})_{i,p} \| \mathcal{P}_E(\mathbf{e}_i) \| \mathbf{m}_p \right], \tag{9}$$

where $\|$ denotes feature concatenation. Separate MLPs f_A and f_R produce feature-wise sigmoid gates:

$$\mathbf{G}_{A,i,p} = \sigma(f_A(\boldsymbol{\xi}_{i,p}(\mathbf{U}))), \qquad \mathbf{G}_{R,i,p} = \sigma(f_R(\boldsymbol{\xi}_{i,p}(\mathbf{V}))). \tag{10}$$

The gates weight the original residuals before temporal convolution:

$$\widetilde{\mathbf{A}} = \mathcal{C}_A(\mathbf{G}_A \odot \mathbf{U}), \qquad \mathbf{A} = \mathrm{LN}_A(\widetilde{\mathbf{A}}), \qquad \widetilde{\mathbf{R}} = \mathcal{C}_R(\mathbf{G}_R \odot \mathbf{V}), \qquad \mathbf{R} = \mathrm{LN}_R(\widetilde{\mathbf{R}}), \tag{11}$$

where $\odot$ denotes element-wise multiplication. The convolutions $\mathcal{C}_A$ and $\mathcal{C}_R$ use the same configuration as $\mathcal{C}_B$ with independent parameters.

Shock uses separate scalar gates for the two views:

$$g^S_{Q,i,p} = \sigma\left(f_{S,Q}\left(\left[|\mathbf{Q}_{i,p}| \| |(\Delta \mathbf{Q})_{i,p}| \| |(\Delta^2 \mathbf{Q})_{i,p}| \right] \right) + c_Q(\mathbf{e}_i, \mathbf{m}_p) \right), \tag{12}$$

where $f_{S,Q}$ scores residual magnitudes and c_Q provides a learned contextual correction. Each c_Q separately projects the node and calendar embeddings, concatenates them, and applies an MLP followed by a scalar projection initialized to zero. The gate MLPs use hidden width $2D$, GELU activation, and dropout. Collecting the scalar gates into $\mathbf{g}^S_U$ and $\mathbf{g}^S_V$, shock is

$$\widetilde{\mathbf{S}} = \mathbf{g}^S_U \odot \mathcal{P}_U(\mathbf{U}) + \mathbf{g}^S_V \odot \mathcal{P}_V(\mathbf{V}), \qquad \mathbf{S} = \mathrm{LN}_S(\widetilde{\mathbf{S}}), \tag{13}$$

where $\mathcal{P}_U$ and $\mathcal{P}_V$ are independent affine feature projections. Scalar gates are broadcast across features and need not sum to one. Magnitudes determine selection, while the projected residuals retain signed information.

Historical integration and temporal projection. An affine map $\mathcal{P}_H : \mathbb{R}^{4D} \to \mathbb{R}^D$ integrates the four components through a residual connection:

$$\mathbf{H}^{(0)} = \mathrm{LN}_H(\mathbf{Z} + \alpha \mathcal{P}_H([\mathbf{B} \| \mathbf{A} \| \mathbf{R} \| \mathbf{S}])), \tag{14}$$

where α is a learned scalar initialized to 0.1 and $\mathbf{H}^{(0)} \in \mathbb{R}^{N \times P \times D}$. The background is also retained separately for functional relation learning. Each component has independent affine temporal and

feature projections. For label $c \in \{B, A, R, S\}$, let $\mathbf{C}^B = \mathbf{B}$, $\mathbf{C}^A = \mathbf{A}$, $\mathbf{C}^R = \mathbf{R}$, and $\mathbf{C}^S = \mathbf{S}$. The temporal map $\mathcal{T}_c$ projects P historical positions to P_{out} target patches, and $\mathcal{F}_c$ maps D features to the forecast dimension D_{f}. Writing $\mathbf{F}_c = \mathcal{F}_c(\mathcal{T}_c(\mathbf{C}^c))$, we obtain

$$\mathbf{F}^{(0)} = \mathrm{LN}_F(\mathcal{P}_F([\mathbf{F}_B \| \mathbf{F}_A \| \mathbf{F}_R \| \mathbf{F}_S]) + \mathbf{M}_{\text{out}}), \tag{15}$$

where $\mathcal{P}_F$ maps $4D_{\text{f}}$ features to D_{f}, giving $\mathbf{F}^{(0)} \in \mathbb{R}^{N \times P_{\text{out}} \times D_{\text{f}}}$. Unlike H, which counts forecasting time steps, P_{out} counts target patches. The calendar representation $\mathbf{M}_{\text{out}} \in \mathbb{R}^{P_{\text{out}} \times D_{\text{f}}}$ uses the shared calendar embedding tables and a separate projection network, and is broadcast across nodes. Background smoothness and shock-gate sparsity are specified in Appendix A.4. The components encode learned evolution roles without imposing exact additive reconstruction or physical conservation.

A.2 Decomposition-Conditioned Dual-Graph Learning

We detail the two relational branches in Section 3.3, omitting the batch dimension. Matrix entry (i, j) represents information received by node i from node j.

Physical propagation graph. Let $\mathbf{J} \in \mathbb{R}_{\geq 0}^{N \times N}$ denote the adjacency of $\mathcal{G}_0$, where $J_{ij} > 0$ indicates a physical edge from i to j. Unlike $\mathbf{J}$, propagation operators use the receiving-node convention: entry (i, j) weights information from node j to node i. The directional supports are

$$\mathbf{T}^{\rightarrow} = \mathrm{RowNorm}(\mathbf{J})^\top, \qquad \mathbf{T}^{\leftarrow} = \mathrm{RowNorm}(\mathbf{J}^\top)^\top, \tag{16}$$

where RowNorm preserves zero rows. For direction d, let $\mathbf{e}^d_{i,\text{rec}}$ and $\mathbf{e}^d_{i,\text{src}} \in \mathbb{R}^{D_e}$ denote receiving and source embeddings. Two learned embedding tables supply these roles, which are exchanged between directions. A shared query projection $\mathbf{W}_q$ and offset-specific key projections $\mathbf{W}_{k,\tau}$ define

$$\omega^{d,\tau}_{ij} = \mathrm{softmax}_j\left(\frac{(\mathbf{W}_q \mathbf{e}^d_{i,\text{rec}})^\top (\mathbf{W}_{k,\tau} \mathbf{e}^d_{j,\text{src}})}{\sqrt{D_e}}\right), \qquad \tau = 1, \ldots, K_t. \tag{17}$$

These affinities vary across edges and temporal positions but are shared across samples.

Sample-dependent affinities use a descriptor $\mathbf{z}^d_i$ formed by concatenating an MLP encoding of $\mathbf{X}_{t-L+1:t,i}$, calendar embeddings at time t, and $\mathbf{e}^d_{i,\text{src}}$. Learned query and key maps $\mathcal{Q}_a$ and $\mathcal{K}_a$ produce D-dimensional features:

$$a^d_{ij} = \mathrm{softmax}_j\left(\frac{\mathcal{Q}_a(\mathbf{z}^d_i)^\top \mathcal{K}_a(\mathbf{z}^d_j)}{\sqrt{D}}\right). \tag{18}$$

The affinities $\mathbf{a}^d$ are shared across temporal positions within each sample. Equation 2 combines them with $\mathbf{T}^d$ and $\boldsymbol{\omega}^{d,\tau}$. Normalization is performed over source nodes separately for each direction and offset, with a small positive denominator floor for numerical stability. The resulting operators are computed once per observed window and reused during forecasting.

To construct propagation states, each directional path applies a scalar node-calendar gate to $\mathbf{H}^{(0)}$, giving $\widetilde{\mathbf{H}}^d$. A GRU followed by causal attention with positional encoding produces the temporal representation $\mathbf{L}^d$. A learned backcast map $\mathcal{P}^d_{\text{back}}$ then defines the propagation residual

$$\mathbf{E}^d = \mathrm{LN}_d\Big(\widetilde{\mathbf{H}}^d - \mathrm{ReLU}(\mathcal{P}^d_{\text{back}}(\mathbf{L}^d))\Big). \tag{19}$$

For spatial order $m = 1, \ldots, K_s$, where K_s is the maximum spatial order, we obtain $\mathbf{W}^{d,\tau}_{[m]}$ by taking the matrix power $(\mathbf{W}^{d,\tau})^m$ and setting its diagonal to zero. Powers are computed from the original operator before diagonal removal, without subsequent renormalization.

For a window ending at patch p, the latest K_t propagation states from $\mathbf{E}^d$ are concatenated along the feature dimension, transformed by a learned linear map and ReLU, and reshaped into $\boldsymbol{\Psi}^d_{j,p,\tau} \in \mathbb{R}^D$. Temporal positions are ordered from oldest to newest; short histories are padded by repeating their

earliest state. The transformation mixes information within the window, so τ indexes a transformed temporal position. The aggregation at spatial order m is

$$\mathbf{g}_{i,p}^{d,[m]} = \sum_{\tau=1}^{K_t} \sum_{j=1}^{N} [\mathbf{W}_{[m]}^{d,\tau}]_{ij} \mathbf{\Psi}_{j,p,\tau}^{d}, \qquad \mathbf{g}_{i,p}^{d,[1]} = \mathbf{g}_{i,p}^{d}, \tag{20}$$

where the first-order case matches Equation 2. The complete graph output combines all spatial orders with a local temporal summary:

$$\mathbf{o}_{i,p}^{d} = \mathcal{P}_g^d \left(\left[\frac{1}{K_t} \sum_{\tau=1}^{K_t} \mathbf{\Psi}_{i,p,\tau}^{d} \;\middle\|\; \mathbf{g}_{i,p}^{d,[1]} \| \cdots \| \mathbf{g}_{i,p}^{d,[K_s]} \right] \right), \tag{21}$$

where $\mathcal{P}_g^d$ is an affine projection followed by dropout. The local summary retains same-node information, while the graph terms exclude diagonal contributions.

Each path predicts the next latent state by adding $\mathbf{o}_{i,p}^{d}$ to $\mathbf{L}_{i,p}^{d}$. The mean of the forward and reverse predictions is fed back to both paths to update their temporal and propagation states. Their predictions are retained separately and projected to D_{f} dimensions, yielding $\mathbf{F}^{\rightarrow}$ and $\mathbf{F}^{\leftarrow}$ on the target-patch grid. Reverse support provides predictive context instead of asserting reverse physical transport.

Functional relation graph. For each node, we summarize $\mathbf{B}_{i,1:P}$ using its mean level, first-to-last change, most recent increment, mean increment, and the final state of a GRU over the background trajectory. An MLP and layer normalization map these summaries to a base descriptor $\overline{\mathbf{h}}_i^B \in \mathbb{R}^{D_b}$, where D_b is the relation descriptor dimension. A separate MLP and layer normalization map the three change summaries to $\mathbf{v}_i^B \in \mathbb{R}^{D_b}$. Change summaries are zero when $P = 1$. Node and calendar context generate feature-wise scale $\boldsymbol{\gamma}_i^B$ and shift $\boldsymbol{\beta}_i^B$, giving

$$\mathbf{h}_i^B = \mathrm{LN}_{B,\mathrm{rel}}\Big((1 + \boldsymbol{\gamma}_i^B) \odot \overline{\mathbf{h}}_i^B + \boldsymbol{\beta}_i^B \Big). \tag{22}$$

The context combines a projected node embedding with pooled calendar embeddings from historical and target patches.

Using these descriptors, Equation 3 computes scores for R_{rel} relation channels, with $\mathbf{M}_r, \mathbf{N}_r \in \mathbb{R}^{D_b \times D_b}$. For receiving node i, let $\mathcal{K}_i$ contain the highest-scoring $\min(k, (N-1)R_{\mathrm{rel}})$ pairs (j, r) with $j \neq i$. The sparse weights are

$$\pi_{ij}^{r} = \begin{cases} \dfrac{\exp(s_{ij}^{r})}{\sum_{(j',r') \in \mathcal{K}_i} \exp(s_{ij'}^{r'})}, & (j,r) \in \mathcal{K}_i, \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

An empty candidate set gives zero weights. Selection is joint over neighbors and relation channels, so multiple channels may connect the same node pair.

Let $\mathcal{V}_r$ denote a relation-specific value map, implemented by a shared feature reduction, a channel-specific projection, and a shared restoration to D dimensions. The historical relational context is

$$\mathbf{C}_{i,p}^{\mathrm{func}} = \sum_{j=1}^{N} \sum_{r=1}^{R_{\mathrm{rel}}} \pi_{ij}^{r} \, \mathcal{V}_r(\mathbf{H}_{j,p}^{(0)}). \tag{24}$$

Thus, background descriptors determine the weights, whereas values retain information from all four components.

Independent affine temporal and feature projections followed by layer normalization map $\mathbf{C}^{\mathrm{func}}$ to a target-patch context. This context is combined with $\mathbf{F}^{(0)}$, $\mathbf{h}^B$, node embeddings, and target calendar features through a shared MLP. Two output heads produce a forecast correction $\boldsymbol{\delta}_{\mathrm{func}}$ and a sigmoid gate $\mathbf{\Lambda}_{\mathrm{func}}$:

$$\mathbf{F}^{\mathrm{func}} = \mathrm{LN}_{\mathrm{func}}\Big(\mathbf{F}^{(0)} + \alpha_{\mathrm{func}} \, \mathbf{\Lambda}_{\mathrm{func}} \odot \boldsymbol{\delta}_{\mathrm{func}} \Big), \tag{25}$$

where α_{func} controls the residual correction scale. The three outputs $\mathbf{F}^{\mathrm{func}}$, $\mathbf{F}^{\rightarrow}$, and $\mathbf{F}^{\leftarrow}$ are passed to adaptive fusion.

A.3 Context-Adaptive Forecast Fusion

We detail the context construction, fusion weights, and output recovery in Section 3.4.

Context construction and fusion. For node i and target patch q, the fusion context is

$$\mathbf{c}_{i,q} = \left[\mathbf{e}_i \| \mathbf{u}_q^{\mathrm{tod}} \| \mathbf{u}_q^{\mathrm{dow}} \| \mathbf{p}_q\right] \in \mathbb{R}^{4D}, \tag{26}$$

where $\mathbf{e}_i \in \mathbb{R}^D$ is the shared node embedding, and $\mathbf{u}_q^{\mathrm{tod}}$ and $\mathbf{u}_q^{\mathrm{dow}}$ are fusion-specific calendar embeddings at the target-patch start time. Their lookup tables contain 288 time-of-day bins and seven day-of-week categories, respectively. The positional encoding $\mathbf{p}_q \in \mathbb{R}^D$ uses the zero-based target-patch position $q-1$:

$$[\mathbf{p}_q]_{2m} = \sin\left(\frac{q-1}{10000^{2m/D}}\right), \qquad [\mathbf{p}_q]_{2m+1} = \cos\left(\frac{q-1}{10000^{2m/D}}\right), \tag{27}$$

for nonnegative integers m whose feature indices are below D. Calendar embeddings describe the target time, whereas $\mathbf{p}_q$ identifies its position within the forecast.

The fusion MLP contains two affine layers with GELU activation and dropout between them. Its three outputs are normalized by softmax and applied to the branch representations as specified in Equation 4. Each weight is a scalar for one node, target patch, and branch, broadcast across D_{f} features.

Prediction and overlap recovery. A shared two-layer prediction head maps each fused representation to a length-ℓ output patch:

$$\hat{\mathbf{y}}_{i,q}^{\mathrm{patch}} = \mathbf{W}_2\, \mathrm{ReLU}\big(\mathbf{W}_1\, \mathrm{ReLU}(\mathbf{F}_{i,q}^{\mathrm{fuse}}) + \mathbf{b}_1\big) + \mathbf{b}_2, \tag{28}$$

where $\hat{\mathbf{y}}_{i,q}^{\mathrm{patch}} \in \mathbb{R}^{\ell}$, $\mathbf{W}_1 \in \mathbb{R}^{D_o \times D_{\mathrm{f}}}$, $\mathbf{W}_2 \in \mathbb{R}^{\ell \times D_o}$, and D_o is the prediction-head hidden dimension. The vectors $\mathbf{b}_1$ and $\mathbf{b}_2$ are biases. Target patches continue the historical patch grid. Using zero-based positions relative to the input-window start, the start of target patch q is $\kappa_q = (P+q-1)s$, with patches included while $\kappa_q < L + H$. For forecasting step $h \in \{1, \ldots, H\}$, define

$$\mathcal{I}_h = \{q : 0 \le L + h - 1 - \kappa_q < \ell\}$$

as the set of target patches covering that step. The final forecast averages their predictions:

$$[\hat{\mathbf{Y}}_{t+1:t+H}]_{h,i} = \frac{\sum_{q \in \mathcal{I}_h} [\hat{\mathbf{y}}_{i,q}^{\mathrm{patch}}]_{L+h-\kappa_q}}{\max(1, |\mathcal{I}_h|)}. \tag{29}$$

Only positions within the forecasting interval are retained, yielding $\hat{\mathbf{Y}}_{t+1:t+H} \in \mathbb{R}^{H \times N}$.

A.4 Learning Objectives

Reconstruction and role consistency. Equation 5 uses component outputs before layer normalization. The smooth-L_1 loss averages the element-wise penalty $\phi(u) = u^2/2$ for $|u| < 1$ and $\phi(u) = |u| - 1/2$ otherwise. The shifted reconstruction is evaluated only where an observed successor exists and is zero when $P = 1$. Gradients through the target $\mathbf{Z}$ are stopped; the component outputs remain differentiable.

For direction d, let $\mathbf{L}_{i,p}^d$, $\mathbf{E}_{i,p}^d$, and $\mathbf{o}_{i,p}^d$ denote the temporal state, propagation residual, and complete graph contribution defined in Appendix A.2. We combine the two directions as $\mathbf{L}_{i,p}^{\Sigma} = \mathbf{L}_{i,p}^{\rightarrow} + \mathbf{L}_{i,p}^{\leftarrow}$, with $\mathbf{E}^{\Sigma}$ and $\mathbf{o}^{\Sigma}$ defined analogously. For each component label $c \in \{B, A, R\}$, define

$$\mathrm{sim}_c(\mathbf{x}, \mathbf{y}) = \cos\big(\mathcal{P}_c^{\mathrm{comp}}(\mathbf{x}), \mathcal{P}_c^{\mathrm{phys}}(\mathbf{y})\big), \tag{30}$$

where $\mathcal{P}_c^{\mathrm{comp}}$ and $\mathcal{P}_c^{\mathrm{phys}}$ are independent D-dimensional linear maps initialized to the identity. Role consistency is

$$\begin{aligned} \mathcal{L}_{\mathrm{role}} = & -\rho_B\, \mathbb{E}_{i,p}\, \mathrm{sim}_B(\mathbf{B}_{i,p+1}, \mathbf{L}_{i,p}^{\Sigma}) \\ & -\rho_A\, \mathbb{E}_{i,p}\, \mathrm{sim}_A(\mathbf{A}_{i,p+1}, \mathbf{o}_{i,p}^{\Sigma}) \\ & -\rho_R\, \mathbb{E}_{i,p}\, \mathrm{sim}_R(\mathbf{R}_{i,p}, \mathbf{E}_{i,p}^{\Sigma}), \end{aligned} \tag{31}$$

where ρ_B, ρ_A, and ρ_R are nonnegative weights. The background and accumulation terms use $p = \max(K_t - 1, 1), \ldots, P - 1$; the release term uses $p = 1, \ldots, P$.

The first two terms associate next-patch background and accumulation with the temporal and graph contributions used to predict that patch. The third associates release with the residual available for propagation at the current patch. All comparisons are within the same node, with gradients through both sides. These are learned semantic constraints rather than assertions of physical conservation. Reconstruction and role consistency are activated by $\chi(e) = \mathbb{I}[e \geq e_0]$, where e_0 is the activation epoch. This indicator does not delay the other auxiliary terms.

Prediction supervision and auxiliary regularization. Let $\mathbf{Y}^\star$ denote the ground-truth future. The final forecast is supervised by $\mathcal{L}_{\text{pred}} = \text{MAE}(\hat{\mathbf{Y}}, \mathbf{Y}^\star)$, where MAE averages absolute errors over valid targets. A shared auxiliary prediction head followed by overlap recovery decodes each branch representation into $\hat{\mathbf{Y}}^b$, giving

$$\mathcal{L}_{\text{branch}} = \frac{1}{3} \sum_{b \in \mathcal{B}} \text{MAE}(\hat{\mathbf{Y}}^b, \mathbf{Y}^\star). \tag{32}$$

This supplies a direct prediction signal to each branch before adaptive fusion. The mean historical next-state prediction of the two physical paths is supervised against the refined history:

$$\mathcal{L}_{\text{hist}} = \mathbb{E}_{i,p<P} \frac{1}{D} \left\| \frac{\mathbf{L}^{\Sigma}_{i,p} + \mathbf{o}^{\Sigma}_{i,p}}{2} - \mathbf{H}^{(0)}_{i,p+1} \right\|_2^2. \tag{33}$$

To discourage redundant temporal and propagation representations, we average $\mathbf{L}^d$ and $\mathbf{E}^d$ over historical patches and subtract each resulting vector's mean across features. Denoting these centered vectors by $\overline{\mathbf{L}}^d_i$ and $\overline{\mathbf{E}}^d_i$, we use

$$\mathcal{L}_{\text{sep}} = \frac{1}{2} \sum_{d \in \{\rightarrow, \leftarrow\}} \mathbb{E}_i \left| \cos\left(\overline{\mathbf{L}}^d_i, \overline{\mathbf{E}}^d_i \right) \right|. \tag{34}$$

This penalty acts within each physical direction rather than directly separating the functional and physical branches.

Background smoothness penalizes second temporal differences. Shock sparsity penalizes the mean activity of the scalar gates g^S_U and g^S_V from Appendix A.1:

$$\mathcal{L}_{\text{bg}} = \mathbb{E}_{i,p \geq 3} \frac{\|\mathbf{B}_{i,p} - 2\mathbf{B}_{i,p-1} + \mathbf{B}_{i,p-2}\|_1}{D}, \qquad \mathcal{L}_{\text{shock}} = \mathbb{E}_{i,p} \frac{g^S_{U,i,p} + g^S_{V,i,p}}{2}. \tag{35}$$

The shock penalty encourages selective gate activation, without enforcing exact zeros in the normalized component. These terms form the auxiliary objective

$$\begin{aligned} \mathcal{L}_{\text{aux}} = \lambda_{\text{branch}} \mathcal{L}_{\text{branch}} + \lambda_{\text{hist}} \mathcal{L}_{\text{hist}} + \lambda_{\text{sep}} \mathcal{L}_{\text{sep}} \\ + \lambda_{\text{bg}} \mathcal{L}_{\text{bg}} + \lambda_{\text{shock}} \mathcal{L}_{\text{shock}}, \end{aligned} \tag{36}$$

where all coefficients are nonnegative. The complete training objective is given in Equation 6.

A.5 Tasks and Datasets

Traffic forecasting. METR-LA and PEMS-BAY contain traffic-speed measurements collected every five minutes from 207 and 325 sensors, respectively. Following the standard protocol, we use the preceding 12 observations to predict the next 12 steps and divide each dataset chronologically into 70%/10%/20% training, validation, and test sets. Missing observations are processed according to the protocol of each benchmark. All normalization statistics are estimated exclusively from the training set and then applied to the validation and test sets.

Water-quality forecasting. We collect and curate publicly available hourly dissolved-oxygen observations from the USGS National Water Information System (NWIS)[1] to construct three five-year

[1] https://waterdata.usgs.gov/

datasets: BASIN030501 (Santee), BASIN031300 (Apalachicola), and BASIN180102 (Klamath). Each dataset contains at least ten monitoring stations, has no more than 10% missing observations per station, and includes at least $N - 1$ directed river connections, ensuring consistent temporal coverage and sufficient network connectivity.

Table 3: Statistics of the water-quality datasets.

Dataset	Basin	Period	Sites	Edges	Station-hours	Observed
BASIN030501	Santee	2014–2018	12	17	525,888	98.66%
BASIN031300	Apalachicola	2017–2022	20	28	876,480	96.44%
BASIN180102	Klamath	2020–2025	11	42	482,064	98.15%
Total	–	–	43	87	1,884,432	97.50%

Invalid sentinels, nonnumeric records, infinite values, and negative measurements are treated as missing. After quality control, 1,837,295 of the 1,884,432 station-hour entries remain observed. Missing entries are filled independently within each station by forward filling followed by backward filling, and the same processed series is used by all methods. Each dataset contains 43,824 hourly timestamps and is divided chronologically into 60%/20%/20% training, validation, and test periods. Directed physical graphs are constructed from downstream river-network reachability. An edge $i \to j$ is retained when station j is downstream of station i within 200 river kilometers; self-loops are excluded. River-path distances define the edge lengths and are converted to Gaussian weights when required. Dataset-level statistics are summarized in Table 3. All models observe the preceding 48 hours and predict $H \in \{48, 96, 144, 192, 240\}$ hours.

A.6 Implementation Details

Architecture and optimization. We use the same backbone across all five datasets, with the shared hyperparameters summarized in Table 4. The physical branch uses a two-patch temporal window and two spatial propagation orders. The functional branch retains the top 16 relations per node, with temporal-context conditioning enabled; shock-conditioned modulation is used in the physical branch. All components are jointly trained from scratch, without exogenous forecasting covariates.

Table 4: Hyperparameters shared across all five datasets.

Architecture		Optimization	
Hidden dimension	32	Optimizer	Adam
Forecast dimension	256	Learning rate	2×10^{-3}
Output hidden units	512	Weight decay	10^{-5}
Attention heads	4	Adam ϵ	10^{-8}
Dropout	0.1	Gradient clipping	5.0
Temporal span K_t	2	LR schedule	MultiStep, $\gamma = 0.5$
Spatial orders K_s	2	Early-stopping patience	50 epochs
Relations per node	16	Checkpoint criterion	validation MAE
Fusion hidden units	128	Normalization	global z-score
Initial fusion weights	$(0.5, 0.25, 0.25)$	Initialization	Xavier
Auxiliary coefficients			
Branch λ_{branch}	0.2	Background λ_{bg}	10^{-4}
History λ_{hist}	0.2	Shock λ_{shock}	10^{-4}
Separation λ_{sep}	0.2	Shock threshold τ	0.1

Dataset-specific settings. Table 5 reports the settings that vary across datasets. Traffic forecasting uses 12 observed and 12 target steps, whereas water-quality forecasting uses 48 observed hours and horizons up to 240 hours. Patch geometry is adjusted to these temporal scales while the remaining architecture is unchanged. For water-quality forecasting, the patch length, stride, and curriculum increment increase with the forecast horizon, as shown in Table 6. At 240 hours, the patch length remains 16 rather than increasing to 20, which preserves sufficient temporal resolution for the 48-hour input. All other settings are shared across horizons and basins.

Training and evaluation. The forecast curriculum begins with the prefix specified in Table 5 and extends it by the same number of steps at each curriculum period until the complete horizon is

Table 5: Dataset-specific hyperparameters. Water-quality settings are shown for the 192-hour horizon; the remaining horizons are given in Table 6.

	METR-LA	PEMS-BAY	BASIN030501	BASIN031300	BASIN180102
Nodes	207	325	12	20	11
History length	12	12	48	48	48
Forecast length	12	12	192	192	192
Patch length / stride	4/2	3/3	16/8	16/8	16/8
Relation channels	4	2	4	4	4
Batch size	64	64	128	128	128
Epoch budget	80	50	40	40	40
Warm-up epochs	0	20	0	0	0
Curriculum start / increment	1	1	16	16	16
Curriculum period (epochs)	6	2	3	3	3
LR milestones	$12, 24, \ldots, 72$	$26, 32, \ldots, 50$	$6, 12, \ldots, 36$	$6, 12, \ldots, 36$	$6, 12, \ldots, 36$
Activation epoch e_0	36	10	0	0	0
Reconstruction λ_{rec}	0.1	0.1	0.02	0.02	0.02
Role consistency $(B/A/R)$	0.2	0.2	0.04	0.04	0.04
Loss space	original	original	normalized	normalized	normalized
Loss masking	applied	applied	none	none	none
Parameters	898,566	904,373	876,750	878,030	876,590

Table 6: Water-quality settings by forecast horizon, shared across the three basins.

Forecast horizon (hours)	48	96	144	192	240
Patch length / stride	4/2	8/4	12/6	16/8	16/8
Curriculum start / increment	4	8	12	16	20

supervised. PEMS-BAY additionally uses a 20-epoch full-horizon warm-up before the curriculum begins. Reconstruction and role-consistency losses are activated at epoch e_0; all remaining terms are active throughout training. Experiments are conducted on a single NVIDIA RTX 5090 GPU.

Parameter sensitivity. We examine three structural hyperparameters governing relation learning: the temporal propagation span K_t, maximum spatial order K_s, and functional Top-K value K. Using a fix random seed, we vary one parameter at a time on METR-LA and BASIN180102 at the 192-hour horizon while keeping all other settings fixed. The shared default is $(K_t, K_s, K) = (2, 2, 16)$.

Table 7: One-at-a-time parameter sensitivity on representative traffic and water-quality settings. Gray rows denote the shared default configuration.

Parameter	**Value**	**METR-LA**		**BASIN180102** ($H = 192$)	
		MAE	RMSE	MAE	MSE
K_t	1	2.8549	5.8625	0.7115	1.5168
K_t	2	2.8439	5.8280	0.7151	1.5289
K_t	3	2.8575	5.8632	0.7105	1.5304
K_t	4	2.8612	5.8803	0.7259	1.5625
K_s	1	2.8725	5.9097	0.7173	1.5485
K_s	2	2.8439	5.8280	0.7151	1.5289
K_s	3	2.8565	5.8787	0.7199	1.5553
K	4	2.8584	5.8771	0.7116	1.5299
K	8	2.8530	5.8705	0.7090	1.5191
K	16	2.8439	5.8280	0.7151	1.5289
K	32	2.8492	5.8572	0.7139	1.5317

Table 7 shows limited sensitivity across the tested ranges. On METR-LA, the shared default achieves the lowest MAE and RMSE in all three sweeps, and the largest MAE increase is only 1.01%. On BASIN180102, MAE changes range from -0.85% to $+1.51\%$, while MSE changes remain between -0.79% and $+2.20\%$. A long temporal span ($K_t = 4$) produces the clearest degradation, whereas $K_s = 2$ performs consistently well on both systems. Performance also varies modestly over $K \in \{4, 8, 16, 32\}$, indicating that functional relation selection does not rely on a narrowly tuned sparsity level. Overall, the shared default provides a stable cross-system configuration without dataset-specific tuning.

Computational complexity. We further compare CANDOR with all 15 baselines in terms of model size and inference cost. For runtime profiling, all methods are evaluated in FP32 on the same NVIDIA RTX 5090 using a common batch size of 64. Each model is first warmed up for 20 forward passes, after which inference latency is averaged over 100 forward passes. Trainable parameter counts are obtained from the corresponding model configurations.

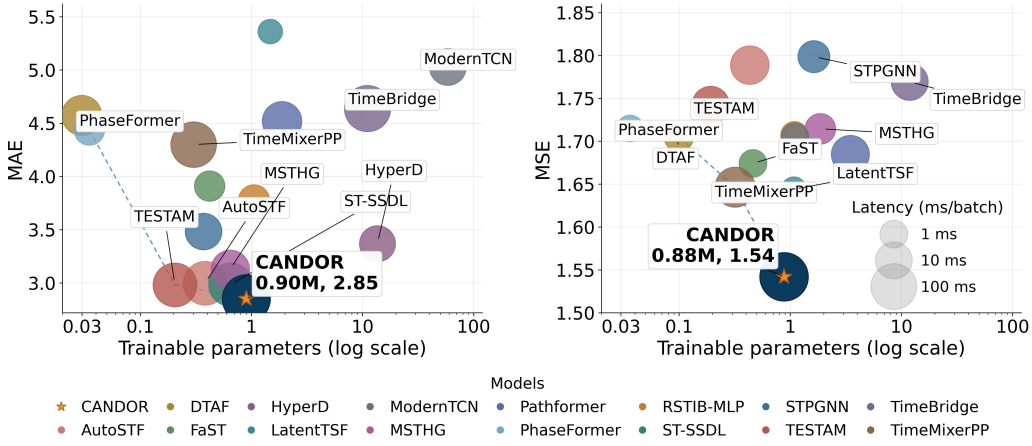


Figure 7: Accuracy-complexity trade-off on representative traffic and water-quality datasets. The horizontal axis shows trainable parameters on a logarithmic scale, the vertical axis reports forecasting error, and bubble area denotes per-batch inference latency. Dashed lines mark the parameter-accuracy Pareto frontier. METR-LA is shown on the left and BASIN180102 on the right.

Figure 7 shows METR-LA and BASIN180102 at the 192-hour horizon as representative traffic and water-quality settings. On METR-LA, CANDOR achieves the lowest MAE of 2.854 with only 0.90M trainable parameters. On BASIN180102, it achieves the lowest MSE of 1.542 with 0.88M parameters. In both cases, CANDOR lies at the high-accuracy end of the parameter-accuracy Pareto frontier. These results show that the forecasting gains of CANDOR do not rely on increasing model size: its decomposition and dual-relation modeling remain compact while providing substantially higher predictive accuracy.

A.7 Compared Methods and Evaluation Protocol

Compared methods. We compare CANDOR with 15 recent forecasting methods from two complementary families. Spatiotemporal models explicitly exploit cross-node interactions, whereas general time-series models provide strong alternatives when spatial propagation is weak or local temporal regularities dominate.

Spatiotemporal forecasting models. TESTAM combines temporal, static-graph, and dynamic-graph experts through context-dependent routing (Lee & Ko, 2024). AutoSTF searches over decoupled spatial and temporal operators and uses multi-patch transfer to capture multi-scale dependencies (Lyu et al., 2025). ST-SSDL models deviations from historical patterns through self-supervised prototype learning and contrastive regularization (Gao et al., 2025). HyperD separates periodic structure from irregular residual dynamics using frequency-aware representations (Shao et al., 2026). RSTIB-MLP distills a graph-based teacher into an efficient MLP under a robust spatiotemporal information bottleneck (Chen et al., 2025). STPGNN identifies pivotal nodes and models their distinct network influence through pivotal graph convolutions (Kong et al., 2024). MSTHG constructs multi-source spatial and temporal hypergraphs to capture dynamic higher-order dependencies (Zhang et al., 2026). FaST combines temporal attention with state-space spatial modeling for efficient long-range dependency learning (Zhao et al., 2026). These methods cover adaptive graphs, architecture search, self-supervised learning, frequency modeling, distillation, pivotal-node reasoning, hypergraphs, and state-space models.

General time-series forecasting models. DTAF addresses temporal and spectral non-stationarity through dual temporal-stabilization and frequency-differencing branches (Lu et al., 2026). La-

tentTSF learns a structured latent representation and performs forecasting within the resulting latent space (Yang et al., 2026). ModernTCN enlarges the effective receptive field of temporal convolutions using large-kernel designs (Luo & Wang, 2024). Pathformer adaptively routes inputs through multi-scale temporal pathways (Chen et al., 2024). PhaseFormer represents temporal patterns through compact phase-wise tokens and lightweight cross-phase routing (Niu et al., 2026). TimeBridge models short-term non-stationarity and long-term cross-variable dependencies through integrated and cointegrated attention (Liu et al., 2025). TimeMixer++ captures temporal patterns across multiple scales and frequency resolutions (Wang et al., 2025). These methods span convolutional, MLP-based, latent-space, frequency-domain, and Transformer-style forecasting paradigms.

Evaluation protocol. All methods mentioned before use identical forecasting targets, input lengths, prediction horizons, chronological splits, and evaluation pipelines, without additional exogenous covariates. Each experiment is repeated using the same three seeds, and the checkpoint with the lowest validation MAE is retained for each run. For traffic forecasting, MAE is the primary metric, with RMSE and MAPE reported as complementary measures. For water-quality forecasting, MSE is the primary metric and MAE is complementary. All metrics are computed after inverse transformation to the original scale.

A.8 Additional Experiments

Tables 8-12 provide complete mean$\pm$std results for all 15 main baselines and CANDOR; the main paper reports the predefined top-eight subsets. These benchmark comparisons use the same forecasting information and do not introduce additional exogenous covariates. CANDOR consistently achieves the best mean performance across both systems and remains consistent across the three seeds. On each traffic dataset, even its worst-seed Overall MAE is lower than the best-seed result of every baseline. For water quality, this strict worst-to-best separation in MSE holds for 222 of 225 basin-horizon-baseline comparisons, with all three exceptions confined to BASIN030501. Thus, CANDOR's advantage is preserved even under its least favorable seeds, demonstrating robust performance across datasets and forecast horizons.

A.9 Ablation Studies

The complete study uses the same nine controlled variants and seed-paired protocol as the main paper, extending the evaluation to all five datasets. Figure 8 reports relative error changes from full CANDOR. The relational ablations reveal a systematic shift across systems. Physical propagation is the dominant relational mechanism in traffic: removing it increases MAE by 9.5% on METR-LA and 2.8% on PEMS-BAY. In river networks, functional relations become more influential. Their removal increases MAE/MSE by $2.5\%/6.5\%$ on BASIN030501, while relation-specific modeling contributes 2.7% in MSE on BASIN031300. On BASIN180102, removing the physical and functional branches increases MSE by 10.5% and 12.8%, respectively, showing that both mechanisms can remain important within the same system. This changing branch importance supports using complementary relation mechanisms instead of a single graph shared across propagation regimes.

The remaining variants validate how these mechanisms are constructed and connected. Both single-direction variants degrade performance where physical propagation is informative, confirming distinct roles for forward propagation and reverse support. Replacing the learned operator with raw adjacency increases METR-LA MAPE by 3.5% and BASIN180102 MSE by 3.9%, showing that topology alone cannot capture temporal-offset and history-conditioned interactions. Removing decomposition increases METR-LA MAE by 1.6% and BASIN180102 MSE by 5.8%, with the largest effect on PEMS-BAY at $4.4\%/6.0\%$ in MAE/MAPE; removing only its auxiliary supervision increases BASIN180102 MAE/MSE by $1.5\%/2.2\%$. Relation-specific modeling and adaptive fusion further contribute up to 3.4% and 2.4% in MSE. Overall, the five-dataset matrix shows that CANDOR's gains do not arise from a single dominant module. Decomposition, direction-aware propagation, non-topological relations, and adaptive fusion contribute under different structural regimes, allowing one architecture to generalize across dense traffic networks and sparse river systems.

Table 8: Complete METR-LA results for all 15 baselines and CANDOR, reported as mean±std over three seeds. Best and second-best means are shown in **bold** and underline, respectively.

	Overall			15 min			30 min			60 min		
Method	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
TESTAM	2.982 ±0.057	6.136 ±0.155	8.14% ±0.21	2.670 ±0.043	5.206 ±0.132	6.89% ±0.18	3.017 ±0.060	6.189 ±0.169	8.25% ±0.24	3.420 ±0.074	7.215 ±0.179	9.86% ±0.24
AutoSTF	2.999 ±0.003	6.195 ±0.002	8.16% ±0.12	2.692 ±0.010	5.231 ±0.022	6.97% ±0.10	3.040 ±0.008	6.254 ±0.013	8.29% ±0.12	3.427 ±0.006	7.292 ±0.019	9.81% ±0.18
ST-SSDL	2.986 ±0.115	6.191 ±0.298	8.13% ±0.28	2.643 ±0.077	5.133 ±0.208	6.80% ±0.23	3.021 ±0.107	6.225 ±0.279	8.23% ±0.27	3.474 ±0.183	7.429 ±0.427	10.00% ±0.40
HyperD	3.370 ±0.006	6.721 ±0.019	10.29% ±0.03	3.091 ±0.004	5.956 ±0.011	9.20% ±0.02	3.451 ±0.007	6.832 ±0.018	10.57% ±0.05	3.778 ±0.007	7.747 ±0.030	12.04% ±0.04
RSTIB-MLP	3.778 ±0.026	9.082 ±0.077	10.12% ±0.05	3.221 ±0.008	7.434 ±0.039	8.36% ±0.04	3.856 ±0.023	9.265 ±0.063	10.39% ±0.05	4.522 ±0.051	10.836 ±0.128	12.41% ±0.10
STPGNN	3.485 ±0.068	6.888 ±0.128	10.12% ±0.38	2.977 ±0.045	5.706 ±0.112	8.12% ±0.31	3.498 ±0.073	6.911 ±0.136	10.18% ±0.43	4.257 ±0.104	8.339 ±0.173	13.08% ±0.52
MSTHG	3.129 ±0.006	6.357 ±0.004	8.84% ±0.07	2.790 ±0.005	5.388 ±0.003	7.42% ±0.10	3.177 ±0.004	6.445 ±0.016	9.03% ±0.08	3.602 ±0.014	7.460 ±0.020	10.71% ±0.16
FaST	3.911 ±0.021	8.932 ±0.023	10.38% ±0.05	3.191 ±0.010	6.977 ±0.012	8.26% ±0.01	3.912 ±0.017	8.850 ±0.017	10.45% ±0.04	4.976 ±0.042	11.231 ±0.046	13.40% ±0.10
DTAF	4.576 ±0.029	10.152 ±0.125	11.74% ±0.04	3.558 ±0.002	7.880 ±0.048	8.86% ±0.02	4.544 ±0.030	10.054 ±0.155	11.62% ±0.05	6.088 ±0.059	12.819 ±0.141	16.06% ±0.09
LatentTSF	5.363 ±0.136	9.720 ±0.057	13.57% ±0.22	4.229 ±0.085	7.957 ±0.036	10.44% ±0.18	5.352 ±0.136	9.778 ±0.043	13.54% ±0.24	6.987 ±0.213	11.709 ±0.081	18.10% ±0.31
ModernTCN	5.031 ±0.010	10.548 ±0.009	12.68% ±0.01	3.941 ±0.023	8.221 ±0.031	9.72% ±0.01	5.005 ±0.031	10.449 ±0.041	12.57% ±0.01	6.645 ±0.022	13.276 ±0.029	17.07% ±0.01
Pathformer	4.520 ±0.012	10.226 ±0.012	11.44% ±0.01	3.529 ±0.010	7.889 ±0.006	8.74% ±0.02	4.494 ±0.015	10.186 ±0.027	11.36% ±0.01	5.952 ±0.022	12.798 ±0.030	15.39% ±0.04
PhaseFormer	4.439 ±0.010	9.562 ±0.009	11.82% ±0.03	3.456 ±0.010	7.260 ±0.012	8.89% ±0.03	4.371 ±0.004	9.386 ±0.007	11.59% ±0.05	5.917 ±0.016	12.271 ±0.004	16.18% ±0.03
TimeBridge	4.639 ±0.747	10.025 ±1.137	12.38% ±2.22	3.740 ±0.750	7.941 ±1.367	9.86% ±2.29	4.582 ±0.734	9.864 ±1.135	12.24% ±2.16	5.999 ±0.725	12.522 ±0.959	16.22% ±2.06
TimeMixer++	4.301 ±0.030	9.143 ±0.037	11.24% ±0.15	3.403 ±0.015	7.050 ±0.014	8.71% ±0.08	4.261 ±0.028	9.007 ±0.030	11.18% ±0.13	5.659 ±0.052	11.632 ±0.061	15.02% ±0.26
CANDOR (Ours)	**2.854** ±0.009	**5.860** ±0.028	**7.77%** ±0.03	**2.548** ±0.008	**4.892** ±0.024	**6.47%** ±0.06	**2.889** ±0.011	**5.891** ±0.029	**7.86%** ±0.05	**3.300** ±0.004	**6.991** ±0.041	**9.62%** ±0.05

Table 9: Complete PEMS-BAY results for all 15 baselines and CANDOR, reported as mean±std over three seeds. Best and second-best means are shown in **bold** and underline, respectively.

	Overall			15 min			30 min			60 min		
Method	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
TESTAM	1.530 ±0.004	3.566 ±0.010	3.44% ±0.01	1.297 ±0.004	2.781 ±0.022	2.74% ±0.01	1.593 ±0.005	3.681 ±0.016	3.59% ±0.02	1.849 ±0.004	4.320 ±0.003	4.34% ±0.02
AutoSTF	1.548 ±0.010	3.599 ±0.019	3.44% ±0.02	1.298 ±0.004	2.758 ±0.011	2.72% ±0.02	1.613 ±0.011	3.712 ±0.020	3.61% ±0.03	1.881 ±0.017	4.388 ±0.026	4.36% ±0.03
ST-SSDL	1.579 ±0.006	3.677 ±0.013	3.51% ±0.02	1.304 ±0.003	2.765 ±0.011	2.70% ±0.01	1.642 ±0.006	3.762 ±0.018	3.65% ±0.02	1.946 ±0.010	4.555 ±0.011	4.55% ±0.04
HyperD	1.796 ±0.012	4.275 ±0.042	4.45% ±0.02	1.643 ±0.014	3.889 ±0.046	4.07% ±0.03	1.856 ±0.013	4.363 ±0.048	4.62% ±0.02	2.109 ±0.011	4.948 ±0.026	5.24% ±0.01
RSTIB-MLP	1.600 ±0.001	3.716 ±0.015	3.61% ±0.01	1.348 ±0.003	2.870 ±0.006	2.84% ±0.00	1.665 ±0.001	3.832 ±0.016	3.77% ±0.01	1.940 ±0.004	4.532 ±0.010	4.60% ±0.04
STPGNN	1.997 ±0.109	4.487 ±0.306	4.75% ±0.47	1.457 ±0.026	3.079 ±0.081	3.04% ±0.07	2.016 ±0.089	4.409 ±0.207	4.68% ±0.35	2.754 ±0.221	5.964 ±0.567	7.14% ±1.03
MSTHG	1.586 ±0.005	3.683 ±0.020	3.59% ±0.00	1.323 ±0.004	2.816 ±0.017	2.82% ±0.01	1.651 ±0.005	3.801 ±0.025	3.75% ±0.01	1.935 ±0.005	4.490 ±0.021	4.56% ±0.00
FaST	1.720 ±0.004	4.153 ±0.026	3.81% ±0.00	1.366 ±0.003	2.983 ±0.023	2.86% ±0.00	1.762 ±0.004	4.147 ±0.024	3.92% ±0.00	2.232 ±0.007	5.368 ±0.029	5.13% ±0.01
DTAF	2.083 ±0.007	5.055 ±0.011	4.59% ±0.05	1.508 ±0.010	3.303 ±0.005	3.11% ±0.04	2.077 ±0.007	4.875 ±0.011	4.54% ±0.04	2.944 ±0.004	6.908 ±0.016	6.78% ±0.08
LatentTSF	2.544 ±0.021	5.034 ±0.009	5.93% ±0.04	1.878 ±0.008	3.559 ±0.015	4.11% ±0.04	2.532 ±0.019	4.938 ±0.009	5.89% ±0.04	3.511 ±0.045	6.609 ±0.009	8.53% ±0.04
ModernTCN	2.152 ±0.001	5.088 ±0.001	4.76% ±0.01	1.580 ±0.001	3.396 ±0.001	3.28% ±0.01	2.153 ±0.001	4.918 ±0.002	4.73% ±0.01	3.014 ±0.001	6.907 ±0.002	6.98% ±0.01
Pathformer	1.911 ±0.027	4.597 ±0.060	4.23% ±0.05	1.460 ±0.009	3.220 ±0.015	3.07% ±0.01	1.923 ±0.023	4.504 ±0.045	4.25% ±0.03	2.596 ±0.057	6.108 ±0.110	5.95% ±0.11
PhaseFormer	2.118 ±0.012	4.981 ±0.022	4.79% ±0.06	1.536 ±0.015	3.270 ±0.028	3.23% ±0.08	2.112 ±0.017	4.807 ±0.026	4.72% ±0.10	2.982 ±0.009	6.800 ±0.023	7.09% ±0.05
TimeBridge	1.911 ±0.002	4.581 ±0.001	4.26% ±0.00	1.425 ±0.001	3.095 ±0.001	2.97% ±0.00	1.921 ±0.002	4.475 ±0.001	4.26% ±0.01	2.631 ±0.003	6.136 ±0.002	6.15% ±0.01
TimeMixer++	1.960 ±0.007	4.396 ±0.006	4.34% ±0.02	1.478 ±0.003	3.073 ±0.008	3.09% ±0.03	1.975 ±0.007	4.322 ±0.006	4.37% ±0.02	2.674 ±0.014	5.832 ±0.010	6.14% ±0.02
CANDOR (Ours)	**1.508** ±0.008	**3.500** ±0.029	**3.34%** ±0.04	**1.257** ±0.003	**2.647** ±0.008	**2.59%** ±0.02	**1.565** ±0.009	**3.582** ±0.025	**3.46%** ±0.04	**1.846** ±0.014	**4.308** ±0.049	**4.31%** ±0.07

Table 10: Complete BASIN030501 results for all 15 baselines and CANDOR, reported as mean±std over three seeds. Best and second-best means are shown in **bold** and underline, respectively.

	48		96		144		192		240		Avg	
Method	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TESTAM	0.422 ±0.015	0.428 ±0.009	0.489 ±0.008	0.464 ±0.013	0.612 ±0.058	0.529 ±0.031	0.691 ±0.033	0.573 ±0.022	0.748 ±0.041	0.601 ±0.023	0.592 ±0.028	0.519 ±0.019
AutoSTF	0.291 ±0.003	0.336 ±0.002	0.483 ±0.013	0.452 ±0.002	0.595 ±0.065	0.514 ±0.027	0.678 ±0.018	0.558 ±0.005	0.781 ±0.050	0.608 ±0.015	0.566 ±0.016	0.494 ±0.006
ST-SSDL	0.339 ±0.013	0.377 ±0.006	0.548 ±0.027	0.495 ±0.011	0.700 ±0.038	0.574 ±0.014	0.853 ±0.015	0.639 ±0.014	0.973 ±0.131	0.695 ±0.050	0.683 ±0.026	0.556 ±0.005
HyperD	2.273 ±0.032	1.163 ±0.008	2.305 ±0.024	1.181 ±0.006	2.314 ±0.003	1.187 ±0.003	2.351 ±0.008	1.200 ±0.000	2.347 ±0.036	1.200 ±0.009	2.318 ±0.015	1.186 ±0.004
RSTIB-MLP	0.370 ±0.004	0.409 ±0.004	0.500 ±0.004	0.477 ±0.002	0.608 ±0.005	0.531 ±0.003	0.685 ±0.002	0.569 ±0.001	0.752 ±0.001	0.602 ±0.000	0.583 ±0.002	0.518 ±0.001
STPGNN	0.352 ±0.008	0.382 ±0.005	0.521 ±0.003	0.478 ±0.003	0.657 ±0.011	0.541 ±0.010	0.740 ±0.013	0.593 ±0.004	0.914 ±0.049	0.671 ±0.023	0.637 ±0.012	0.533 ±0.007
MSTHG	0.347 ±0.016	0.359 ±0.003	0.449 ±0.006	0.439 ±0.002	0.558 ±0.007	0.494 ±0.004	0.629 ±0.017	0.536 ±0.007	0.706 ±0.017	0.570 ±0.005	0.538 ±0.009	0.480 ±0.003
FaST	0.307 ±0.002	0.336 ±0.000	0.470 ±0.003	0.429 ±0.001	0.604 ±0.001	0.495 ±0.001	0.711 ±0.002	0.547 ±0.001	0.802 ±0.001	0.589 ±0.000	0.579 ±0.001	0.479 ±0.000
DTAF	0.320 ±0.001	0.344 ±0.000	0.487 ±0.000	0.438 ±0.000	0.623 ±0.002	0.505 ±0.000	0.732 ±0.001	0.557 ±0.000	0.821 ±0.000	0.598 ±0.000	0.597 ±0.001	0.488 ±0.000
LatentTSF	0.321 ±0.000	0.357 ±0.001	0.485 ±0.001	0.456 ±0.001	0.613 ±0.000	0.523 ±0.001	0.715 ±0.002	0.574 ±0.003	0.799 ±0.003	0.616 ±0.002	0.586 ±0.001	0.505 ±0.001
ModernTCN	0.326 ±0.000	0.351 ±0.000	0.498 ±0.000	0.445 ±0.000	0.636 ±0.000	0.512 ±0.000	0.748 ±0.000	0.563 ±0.000	0.839 ±0.000	0.604 ±0.000	0.609 ±0.000	0.495 ±0.000
Pathformer	0.311 ±0.001	0.334 ±0.001	0.477 ±0.001	0.429 ±0.000	0.608 ±0.002	0.495 ±0.000	0.714 ±0.002	0.546 ±0.000	0.807 ±0.003	0.589 ±0.001	0.583 ±0.001	0.479 ±0.000
PhaseFormer	0.317 ±0.002	0.344 ±0.001	0.486 ±0.001	0.441 ±0.001	0.619 ±0.001	0.507 ±0.001	0.728 ±0.001	0.559 ±0.001	0.815 ±0.002	0.599 ±0.001	0.593 ±0.000	0.490 ±0.000
TimeBridge	0.311 ±0.005	0.339 ±0.002	0.479 ±0.010	0.434 ±0.004	0.618 ±0.013	0.502 ±0.003	0.724 ±0.002	0.554 ±0.001	0.819 ±0.030	0.596 ±0.009	0.590 ±0.010	0.485 ±0.003
TimeMixer++	0.313 ±0.000	0.342 ±0.001	0.477 ±0.001	0.436 ±0.001	0.611 ±0.008	0.502 ±0.003	0.719 ±0.010	0.554 ±0.003	0.811 ±0.002	0.598 ±0.001	0.586 ±0.000	0.486 ±0.000
CANDOR (Ours)	**0.289** ±0.002	**0.328** ±0.002	**0.433** ±0.007	**0.418** ±0.003	**0.544** ±0.006	**0.478** ±0.004	**0.611** ±0.009	**0.519** ±0.005	**0.665** ±0.013	**0.548** ±0.005	**0.508** ±0.001	**0.458** ±0.001

Table 11: Complete BASIN031300 results for all 15 baselines and CANDOR, reported as mean±std over three seeds. Best and second-best means are shown in **bold** and underline, respectively.

	48		96		144		192		240		Avg	
Method	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TESTAM	0.315 ±0.007	0.389 ±0.002	0.369 ±0.003	0.425 ±0.003	0.449 ±0.014	0.480 ±0.006	0.515 ±0.006	0.523 ±0.003	0.565 ±0.028	0.552 ±0.013	0.443 ±0.010	0.474 ±0.004
AutoSTF	0.213 ±0.005	0.297 ±0.005	0.372 ±0.021	0.417 ±0.011	0.482 ±0.030	0.489 ±0.016	0.524 ±0.035	0.521 ±0.018	0.599 ±0.035	0.563 ±0.014	0.438 ±0.011	0.458 ±0.004
ST-SSDL	0.244 ±0.010	0.329 ±0.008	0.446 ±0.052	0.462 ±0.028	0.550 ±0.033	0.528 ±0.016	0.652 ±0.063	0.584 ±0.025	0.775 ±0.070	0.644 ±0.032	0.533 ±0.013	0.509 ±0.008
HyperD	1.634 ±0.017	0.959 ±0.005	1.660 ±0.016	0.976 ±0.004	1.670 ±0.008	0.989 ±0.002	1.692 ±0.012	1.002 ±0.005	1.701 ±0.009	1.006 ±0.002	1.671 ±0.004	0.986 ±0.002
RSTIB-MLP	0.314 ±0.008	0.397 ±0.007	0.430 ±0.002	0.470 ±0.002	0.502 ±0.005	0.511 ±0.004	0.557 ±0.002	0.544 ±0.002	0.600 ±0.003	0.568 ±0.001	0.480 ±0.003	0.498 ±0.003
STPGNN	0.326 ±0.019	0.411 ±0.020	0.614 ±0.053	0.592 ±0.030	0.658 ±0.081	0.603 ±0.032	0.734 ±0.077	0.640 ±0.034	0.724 ±0.055	0.639 ±0.033	0.611 ±0.044	0.577 ±0.020
MSTHG	0.227 ±0.004	0.310 ±0.001	0.371 ±0.005	0.417 ±0.004	0.469 ±0.013	0.481 ±0.006	0.529 ±0.004	0.519 ±0.002	0.585 ±0.008	0.552 ±0.006	0.436 ±0.004	0.456 ±0.002
FaST	0.213 ±0.002	0.294 ±0.002	0.350 ±0.001	0.396 ±0.001	0.441 ±0.000	0.454 ±0.001	0.507 ±0.000	0.495 ±0.000	0.557 ±0.001	0.525 ±0.001	0.414 ±0.001	0.433 ±0.001
DTAF	0.237 ±0.001	0.314 ±0.001	0.377 ±0.001	0.414 ±0.001	0.471 ±0.000	0.472 ±0.000	0.539 ±0.000	0.512 ±0.000	0.591 ±0.001	0.542 ±0.000	0.443 ±0.000	0.451 ±0.000
LatentTSF	0.248 ±0.000	0.332 ±0.000	0.381 ±0.000	0.428 ±0.000	0.468 ±0.000	0.482 ±0.000	0.530 ±0.000	0.520 ±0.000	0.578 ±0.000	0.547 ±0.000	0.441 ±0.000	0.462 ±0.000
ModernTCN	0.253 ±0.000	0.330 ±0.000	0.392 ±0.000	0.426 ±0.000	0.486 ±0.000	0.483 ±0.000	0.553 ±0.000	0.521 ±0.000	0.606 ±0.000	0.551 ±0.000	0.458 ±0.000	0.462 ±0.000
Pathformer	0.220 ±0.001	0.298 ±0.001	0.357 ±0.002	0.400 ±0.001	0.451 ±0.001	0.461 ±0.000	0.522 ±0.001	0.504 ±0.001	0.580 ±0.003	0.537 ±0.001	0.426 ±0.001	0.440 ±0.000
PhaseFormer	0.231 ±0.004	0.312 ±0.003	0.369 ±0.001	0.411 ±0.000	0.465 ±0.002	0.470 ±0.002	0.532 ±0.003	0.510 ±0.002	0.584 ±0.003	0.539 ±0.002	0.436 ±0.001	0.448 ±0.001
TimeBridge	0.222 ±0.001	0.301 ±0.001	0.365 ±0.006	0.406 ±0.004	0.468 ±0.012	0.468 ±0.005	0.527 ±0.001	0.505 ±0.001	0.593 ±0.001	0.539 ±0.001	0.435 ±0.003	0.444 ±0.002
TimeMixer++	0.217 ±0.002	0.299 ±0.001	0.354 ±0.002	0.400 ±0.002	0.446 ±0.002	0.460 ±0.001	0.512 ±0.002	0.500 ±0.002	0.559 ±0.002	0.528 ±0.001	0.418 ±0.000	0.437 ±0.000
CANDOR (Ours)	**0.205** ±0.004	**0.287** ±0.003	**0.331** ±0.004	**0.388** ±0.003	**0.420** ±0.006	**0.449** ±0.005	**0.487** ±0.006	**0.493** ±0.004	**0.528** ±0.008	**0.520** ±0.006	**0.394** ±0.005	**0.428** ±0.004

Table 12: Complete BASIN180102 results for all 15 baselines and CANDOR, reported as mean±std over three seeds. Best and second-best means are shown in **bold** and underline, respectively.

	48		96		144		192		240		Avg	
Method	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TESTAM	0.900 ±0.005	0.551 ±0.002	1.215 ±0.104	0.622 ±0.015	1.598 ±0.148	0.733 ±0.024	1.743 ±0.099	0.788 ±0.003	2.066 ±0.244	0.865 ±0.011	1.505 ±0.108	0.712 ±0.008
AutoSTF	0.750 ±0.030	0.455 ±0.008	1.142 ±0.086	0.592 ±0.024	1.489 ±0.094	0.703 ±0.018	1.789 ±0.195	0.783 ±0.043	2.208 ±0.129	0.877 ±0.021	1.476 ±0.032	0.682 ±0.008
ST-SSDL	1.090 ±0.056	0.635 ±0.037	1.704 ±0.228	0.811 ±0.066	2.171 ±0.335	0.956 ±0.127	3.077 ±0.690	1.121 ±0.156	3.701 ±0.406	1.287 ±0.145	2.349 ±0.093	0.962 ±0.032
HyperD	2.956 ±0.024	0.990 ±0.008	3.107 ±0.028	1.047 ±0.005	3.251 ±0.036	1.105 ±0.005	3.447 ±0.075	1.168 ±0.026	3.487 ±0.077	1.183 ±0.020	3.250 ±0.023	1.099 ±0.006
RSTIB-MLP	0.921 ±0.020	0.606 ±0.013	1.214 ±0.019	0.672 ±0.012	1.486 ±0.030	0.743 ±0.017	1.707 ±0.001	0.802 ±0.001	1.882 ±0.014	0.848 ±0.007	1.442 ±0.007	0.734 ±0.005
STPGNN	1.088 ±0.128	0.627 ±0.043	1.508 ±0.201	0.746 ±0.036	1.903 ±0.237	0.839 ±0.056	1.799 ±0.045	0.822 ±0.015	2.103 ±0.128	0.914 ±0.051	1.680 ±0.018	0.790 ±0.014
MSTHG	0.748 ±0.011	0.460 ±0.003	1.085 ±0.013	0.591 ±0.005	1.404 ±0.026	0.688 ±0.002	1.715 ±0.116	0.775 ±0.023	1.912 ±0.045	0.835 ±0.010	1.373 ±0.025	0.670 ±0.003
FaST	0.725 ±0.001	0.441 ±0.001	1.096 ±0.011	0.564 ±0.003	1.390 ±0.005	0.652 ±0.001	1.674 ±0.020	0.732 ±0.004	1.879 ±0.024	0.789 ±0.003	1.353 ±0.002	0.636 ±0.001
DTAF	0.743 ±0.002	0.451 ±0.001	1.111 ±0.005	0.578 ±0.001	1.429 ±0.000	0.672 ±0.000	1.705 ±0.004	0.748 ±0.000	1.924 ±0.002	0.810 ±0.000	1.382 ±0.001	0.652 ±0.000
LatentTSF	0.739 ±0.002	0.473 ±0.001	1.091 ±0.002	0.606 ±0.004	1.394 ±0.007	0.701 ±0.008	1.644 ±0.003	0.772 ±0.000	1.846 ±0.002	0.835 ±0.003	1.343 ±0.002	0.678 ±0.003
ModernTCN	0.748 ±0.000	0.461 ±0.000	1.112 ±0.000	0.585 ±0.000	1.433 ±0.000	0.678 ±0.000	1.705 ±0.000	0.753 ±0.000	1.925 ±0.000	0.814 ±0.000	1.385 ±0.000	0.658 ±0.000
Pathformer	0.732 ±0.003	0.440 ±0.001	1.107 ±0.006	0.563 ±0.002	1.408 ±0.010	0.653 ±0.002	1.685 ±0.014	0.731 ±0.001	1.890 ±0.009	0.791 ±0.001	1.364 ±0.004	0.636 ±0.000
PhaseFormer	0.752 ±0.001	0.455 ±0.000	1.122 ±0.002	0.583 ±0.001	1.440 ±0.003	0.676 ±0.001	1.715 ±0.007	0.753 ±0.001	1.931 ±0.007	0.814 ±0.001	1.392 ±0.002	0.656 ±0.000
TimeBridge	0.745 ±0.013	0.445 ±0.002	1.129 ±0.032	0.572 ±0.005	1.418 ±0.014	0.660 ±0.001	1.770 ±0.029	0.753 ±0.007	1.950 ±0.055	0.803 ±0.007	1.402 ±0.006	0.647 ±0.002
TimeMixer++	0.726 ±0.006	0.443 ±0.001	1.089 ±0.011	0.565 ±0.002	1.394 ±0.005	0.659 ±0.001	1.647 ±0.015	0.731 ±0.005	1.901 ±0.007	0.800 ±0.003	1.351 ±0.002	0.639 ±0.001
CANDOR (Ours)	**0.712** ±0.001	**0.440** ±0.000	**1.040** ±0.004	**0.559** ±0.003	**1.302** ±0.024	**0.644** ±0.005	**1.542** ±0.011	**0.715** ±0.001	**1.763** ±0.027	**0.780** ±0.014	**1.272** ±0.012	**0.628** ±0.004

A.10 Detailed Mechanism Analysis

This section examines whether CANDOR's gains follow from its intended mechanisms beyond forecasting accuracy. We first ground the learned Shock using rainfall unseen by CANDOR, verify that rainfall is forecasting-relevant, and test whether it can supplement or replace the Shock pathway. We then assess whether the decomposition is information-preserving and role-specific, trace release and accumulation through delayed physical propagation, and examine how background-conditioned functional relations complement topology under adaptive fusion. These analyses evaluate the physical interpretability and end-to-end coherence of decomposition-conditioned relation learning.

External grounding of the learned Shock. We examine whether the learned decomposition acquires physical meaning beyond its training objectives. Because rainfall can induce abrupt changes in river conditions, we use observed precipitation as an external probe of the Shock component. CANDOR never observes rainfall during training, model selection, or standard forecasting. After training, we compare station-aligned hourly precipitation from NASA GPM IMERG V07 with the mean Shock-gate activity $(g^S_{U,i,p} + g^S_{V,i,p})/2$, the same quantity regularized by $\mathcal{L}_{\text{shock}}$ in Eq. 35. We compute Spearman correlations over lags from -120 to $+120$ hours at two-hour intervals. Patch-level gate activities are averaged over the patches covering each hour. Node-level correlations are averaged within each seed and forecasting horizon, after which the maximum correlation is selected for each horizon and averaged across horizons. Statistical significance is assessed using 20,000 basin-level circular shifts that preserve rainfall's temporal structure and repeat the complete lag-selection procedure. Table 13 shows significant positive associations in all three basins. The selected correlation is also positive for all 45 seed-horizon checkpoints. Although the magnitudes vary across systems, their consistent excess over the shift-based null provides external physical grounding for the learned Shock role.

Predictive relevance of rainfall. To assess whether rainfall carries predictive information, we evaluate TimeXer (Wang et al., 2024b) and XLinear (Chen et al., 2026), both of which support ex-

Table 13: Association between the learned Shock-gate activity and rainfall unobserved by CANDOR. The statistic is the mean horizon-wise maximum Spearman correlation. The null distribution uses 20,000 circular shifts with the same lag-selection procedure.

Dataset	ρ	Null mean	Excess	p
BASIN030501	0.0347	0.0113	0.0234	0.0068
BASIN031300	0.0445	0.0149	0.0296	$< 5 \times 10^{-5}$
BASIN180102	0.1310	-0.0115	0.1425	0.0181

ogenous inputs, with and without aligned rainfall. We report these auxiliary comparisons separately because the rainfall-assisted models receive an additional input unavailable to CANDOR and the standard baselines. Table 14 shows modest but consistent improvements across all three basins, confirming the predictive relevance of rainfall.

Table 14: TimeXer and XLinear with and without rainfall input. Results are averaged over three seeds and five forecasting horizons.

		BASIN030501		BASIN031300		BASIN180102	
Method	**Rainfall**	MAE	MSE	MAE	MSE	MAE	MSE
TimeXer	No	0.4850	0.5883	0.4415	0.4280	0.6506	1.3794
TimeXer	Yes	**0.4835**	**0.5855**	**0.4379**	**0.4231**	**0.6493**	**1.3764**
XLinear	No	0.4857	0.5926	0.4428	0.4333	0.6525	1.3902
XLinear	Yes	**0.4855**	**0.5923**	**0.4422**	**0.4323**	**0.6524**	**1.3899**

Intervening on the Shock pathway. We finally test whether rainfall adds to, or can replace, the learned Shock representation. At the 192-hour horizon, *RainAdd* injects aligned rainfall into the native Shock stream, whereas *RainOnly* replaces that stream with rainfall. Both variants follow the same configuration and the same three seeds as full CANDOR. Table 15 shows that adding rainfall yields no distinguishable improvement, indicating that its predictive contribution substantially overlaps with the learned Shock representation. Rainfall alone provides no consistent advantage and degrades MSE on BASIN180102 beyond the paired-variability criterion. These analyses show that CANDOR recovers rainfall-related disturbances without observing rainfall, while the learned Shock retains predictive information beyond precipitation alone.

Table 15: Rainfall interventions at the 192-hour horizon. Values are seed-paired relative changes (%) from full CANDOR.

	RainAdd		RainOnly	
Dataset	ΔMAE	ΔMSE	ΔMAE	ΔMSE
BASIN030501	-0.33	-1.05	-0.78	-1.04
BASIN031300	-1.15	-1.86	-0.67	-0.65
BASIN180102	+0.50	+0.49	-0.16	+1.77†

†Change exceeds the paired-variability criterion in Appendix A.9.

Reference-consistent decomposition. Figure 5(a) evaluates two complementary properties of the learned decomposition: reconstruction sufficiency and role differentiation. The current-reference reconstruction

$$\mathbf{Z}_{i,p} \approx \widetilde{\mathbf{B}}_{i,p} + \widetilde{\mathbf{A}}_{i,p} + \widetilde{\mathbf{S}}_{i,p} \tag{37}$$

explains the observed state relative to its current background, making $\widetilde{\mathbf{A}}_{i,p}$ responsible for gradual buildup above that reference. The shifted-reference reconstruction

$$\mathbf{Z}_{i,p} \approx \widetilde{\mathbf{B}}_{i,\min(p+1,P)} + \widetilde{\mathbf{R}}_{i,p} + \widetilde{\mathbf{S}}_{i,p} \tag{38}$$

instead measures the state relative to the subsequent background, providing a complementary view of relaxation toward the next reference. The shock component is shared by both reconstructions so that abrupt deviations need not be absorbed into the smoother accumulation or release components.

The agreement between the reconstructed and observed trajectories shows that decomposition preserves information required to represent the local dynamics. Meanwhile, the differentiated component regions in the shared representation space indicate that reconstruction is not achieved by four interchangeable copies of the same signal. These two diagnostics provide complementary evidence: reconstruction assesses whether the components remain collectively sufficient, whereas representation separation assesses whether they acquire specialized roles.

Delayed physical propagation and role consistency. For visualization, we combine the two directional propagation paths as

$$\mathbf{E}^{\Sigma} = \mathbf{E}^{\rightarrow} + \mathbf{E}^{\leftarrow}, \qquad \mathbf{o}^{\Sigma} = \mathbf{o}^{\rightarrow} + \mathbf{o}^{\leftarrow}, \tag{39}$$

where $\mathbf{E}^{\Sigma}$ is the propagation residual and $\mathbf{o}^{\Sigma}$ is the resulting incoming graph contribution. Figure 5(b) examines their temporal ordering. Under the two-patch propagation window used by CANDOR, the localized states $\mathbf{E}^{\Sigma}(t)$ and $\mathbf{E}^{\Sigma}(t+1)$ map to the later graph contribution $\mathbf{o}^{\Sigma}(t+2)$. This ordering is consistent with delayed transport: a deviation must first form a propagating state before its effect appears as incoming information at another node.

Figure 5(c) examines whether this transition preserves the intended component semantics. Release $\mathbf{R}_{i,p}$ aligns with the contemporaneous propagation residual $\mathbf{E}^{\Sigma}_{i,p}$, while subsequent accumulation $\mathbf{A}_{i,p+1}$ aligns with the incoming contribution $\mathbf{o}^{\Sigma}_{i,p}$. Thus, release-associated deviations provide states that can be transmitted, whereas the received graph contribution supports later buildup at the receiving node. These alignments are learned representation-level correspondences rather than assertions of exact mass conservation. Likewise, the reverse-support path provides counter-directional predictive context and should not be interpreted as backward physical transport.

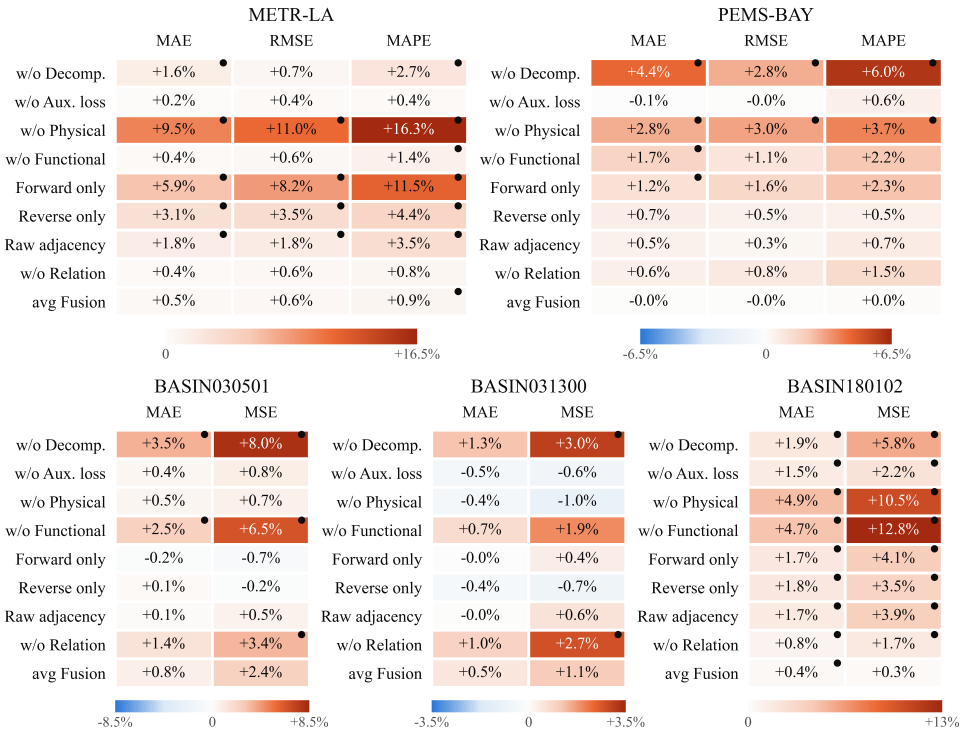


Figure 8: Ablation variants across all five datasets. Values are seed-paired relative changes (%) from full CANDOR; higher is worse. Dots denote mean changes exceeding twice the full-model seed variation.

Background-conditioned relations beyond topology. Figure 6(a) compares the directed physical support $\mathcal{G}_0$ with the learned functional relations for a representative water-quality site. Several nodes

outside the physical neighborhood receive substantial functional weights, showing that predictive dependence is not restricted to direct river connectivity. Such links can arise when distant sites share persistent temporal behavior even though no direct transport edge connects them.

Importantly, these relations are inferred from background dynamics $\mathbf{B}$ rather than from unrestricted mixtures of all components. Persistent background behavior provides a more stable basis for relation discovery, while transient accumulation, release, and shock events are prevented from directly redefining the functional graph. A functional edge should therefore be interpreted as a predictive relationship, not as evidence of an unobserved physical connection or causal transport pathway. Its role is to complement, rather than replace, the physically supported graph.

Context-adaptive use of relation mechanisms. Figure 6(b) reports both the mean allocation and the 10th-90th percentile range of the fusion weights across nodes, samples, and forecast positions. The nonzero ranges show that fusion is not merely dataset-specific: the relative use of functional, forward, and reverse-support forecasts also changes within each dataset and across forecasting contexts. The allocation patterns are consistent with the different propagation regimes. METR-LA assigns greater average weight to physical support, where directed interactions provide useful information about congestion propagation. PEMS-BAY and the water-quality datasets assign relatively more weight to functional forecasts, indicating that persistent dependencies beyond immediate topology become more useful when physical coupling is weaker, sparser, or subject to heterogeneous delays. No single branch dominates uniformly. Together with the nonlocal relations in panel (a), this result supports the central design of CANDOR: physical propagation and functional dependence capture distinct sources of predictive information, and their relevance must be adapted to the observed system and forecasting context.