REASONING SELECTIVELY OVER FORECAST ERRORS WITH EVIDENCE-CONDITIONED PROBABILISTIC CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

A probabilistic forecast can be well calibrated on average yet remain unreliable under a particular structural state. We propose **evidence-guided selective probabilistic reasoning**, a framework that separates the choice of a residual condition from the estimation and deployment of its numerical correction. A numerical forecaster provides a predictive center and asymmetric interval. A selective multi-agent workflow examines bounded structural evidence and produces a residual *ConditionSpec*, which deterministic estimators use to retrieve mature residuals and compute center and tail corrections. An outcome-grounded Gate then selects how much of the candidate to deploy using comparable historical candidates frozen before their outcomes were known. Delayed outcomes populate separate views for residual estimation, workflow allocation, and deployment. Experiments on XuanCheng and PEMS-BAY show dataset-dependent effects of evidence conditioning and an explicit accuracy–computation trade-off. Selective reasoning reduces live LLM calls by 38.7% and 60.2% relative to Always FULL, respectively, with higher MAE than Always FULL but lower point-prediction errors than the Base. On PEMS-BAY, suppressed candidates have negative mean interval-score gain, whereas fully deployed candidates have positive mean gain. These results connect structural reasoning to measurable candidate quality and deployment behavior, providing a traceable way to control both reasoning effort and forecast correction.

1 INTRODUCTION

Modern time-series forecasting has progressed from lightweight linear predictors to graph neural networks and large pretrained temporal models, substantially improving the ability to capture temporal and cross-variable dependencies (Zeng et al., 2023; Wu et al., 2020; Das et al., 2024; Ansari et al., 2024). Probabilistic calibration further complements point forecasts with uncertainty estimates and can adapt to changing error statistics (Gibbs & Candès, 2021). Yet a forecast that performs well or is well calibrated on average can still be unreliable under a particular structural state. In graph-structured systems, the dependencies active at a given forecast origin may differ from those represented by the historical residual pool: upstream disturbances may become unusually strong, several dependency paths may become simultaneously active, or structurally conflicting signals may appear. Such conditions do not necessarily require a new forecasting model, but they raise a more specific question: *is the current Base forecast conditionally adequate for the evidence available now?* Large language models (LLMs) and multi-agent systems provide a promising mechanism for inspecting heterogeneous structural evidence and reasoning over such case-specific conditions (Yao et al., 2023; Wu et al., 2024; Hong et al., 2024). However, a semantically plausible explanation does not establish whether the Base forecast retains residual bias, how far its predictive center should move, how either tail of its prediction interval should change, or whether a proposed correction will ultimately be beneficial. The missing capability is therefore not simply another numerical forecaster or a more elaborate reasoning chain, but a mechanism that can **selectively recognize when additional reasoning is warranted, translate structural evidence into statistically grounded probabilistic correction, and abstain when that correction is insufficiently supported**.

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.

We address this problem through **evidence-guided selective probabilistic reasoning**, built around a deliberate separation of three decisions that are often conflated: *whether to reason, what residual condition the current evidence supports, and how much of the resulting correction should be deployed*. A lightweight numerical predictor with rolling asymmetric residual calibration first provides a reliable Base forecast. Forecast-time structural dependencies and already-matured residual states are then organized into bounded evidence rather than directly converted into correction magnitudes. When additional reasoning is justified, a finite multi-agent workflow inspects this evidence and converts it into a structured *ConditionSpec* describing which aspects of the historical residual process are relevant to the current state. The key separation is that the agents determine *which condition should be examined*, whereas deterministic statistical modules determine *how much numerical correction that condition supports*: compatible mature Base residuals are retrieved to estimate the conditional center bias and asymmetric tail behavior, yielding a traceable candidate probabilistic forecast. Even this candidate is not automatically accepted. A separate outcome-grounded Gate compares it with historically similar candidates that were themselves frozen before their outcomes became known and determines the deployment strength of the current correction. Thus, structural evidence never directly determines a numerical shift, agent agreement is not treated as numerical confidence, and reasoning does not decide how strongly its own proposal should be trusted. The resulting system can abstain twice—before unnecessary reasoning and again before deploying an unsupported correction—while delayed outcomes continually provide genuine evidence about which reasoning workflows and candidate corrections have been useful.

This formulation leads to three main contributions. **First, we introduce evidence-conditioned probabilistic correction that uses structured reasoning to identify forecast-specific residual conditions and deterministic statistics to estimate both predictive-center and asymmetric interval corrections from mature historical evidence.** This converts the role of LLM reasoning from unconstrained numerical prediction to an interpretable interface between structural evidence and conditional residual estimation. **Second, we decouple reasoning allocation from correction acceptance**: reasoning depth is selected before a candidate exists, whereas deployment is decided only after the actual candidate is available and can be compared with historically frozen candidates, enabling selective computation as well as selective correction. **Third, we develop an outcome-grounded feedback protocol in which forecast states, candidates, and deployment decisions are fixed before ground truth becomes available, while delayed outcomes populate separate historical views for residual estimation, workflow allocation, and candidate deployment.** This separation reduces selection and hindsight leakage and makes every numerical correction traceable to mature residual records and every adaptive acceptance decision auditable against realized outcomes. Together, these components provide a general framework for turning LLM-based temporal reasoning into a statistically grounded and selectively deployable correction mechanism rather than a replacement for the underlying numerical forecaster.

2 Related Work

2.1 Numerical Forecasting

Numerical forecasting ranges from regularized autoregression, including Ridge-based predictors, to neural models of multivariate dependencies. LTSF-Linear shows that simple linear mappings can remain competitive (Zeng et al., 2023), while MTGNN jointly models graph structure and temporal dynamics (Wu et al., 2020). TimesFM and Chronos further extend forecasting through large-scale pretraining on diverse sequences (Das et al., 2024; Ansari et al., 2024). Beyond point prediction, adaptive conformal inference updates prediction sets using observed errors to track coverage under distribution shift (Gibbs & Candès, 2021). These methods provide efficient numerical prediction, dependency modeling, and statistical calibration. However, they do not by themselves determine how forecast-conditioned structured evidence should be used to assess a case-specific correction, while aggregate calibration may not reflect correction reliability under particular operating states.

2.2 LLMs and Multi-Agent Reasoning

LLMs have increasingly been used for temporal modeling and explicit forecasting reasoning. Time-LLM reprograms numerical sequences into a pretrained language model (Jin et al., 2024), whereas Time-R1 trains multi-step forecasting reasoning through supervised and reinforcement fine-

tuning (Zhou et al., 2026). ReAct interleaves reasoning with external actions (Yao et al., 2023). Multi-agent frameworks further organize collaborative reasoning: AutoGen supports interactions among models, tools, and humans (Wu et al., 2024), MetaGPT coordinates specialized roles through structured operating procedures (Hong et al., 2024), and multi-agent debate improves reasoning through iterative exchange (Du et al., 2024). These approaches offer flexible contextual reasoning, tool use, and structured collaboration. However, reasoning coherence or agent agreement alone does not establish whether a generated numerical correction is beneficial, nor how strongly such a correction should be deployed for a particular prediction case.

3 METHODOLOGY

3.1 PROBLEM FORMULATION

For a target series v represented as a node in a graph-structured multivariate time series, we predict $y_{v,t+\Delta}$ from forecast origin t. Rather than replacing the numerical forecasting backbone, we perform **selective evidence-guided correction** of its probabilistic output. The Base forecast is $B_t^0 = (\mu_t^0, L_t^0, U_t^0)$, where $[L_t^0, U_t^0]$ has nominal coverage $1-\alpha$. When current structural evidence indicates that the residual behavior underlying B_t^0 may be unrepresentative, an evidence-conditioned candidate B_t^R is constructed and subsequently deployed only to a data-supported degree:

$$B_t^0 \xrightarrow{\text{evidence-conditioned reasoning}} B_t^R \xrightarrow{\text{outcome-grounded deployment}} B_t^{\text{final}}. \tag{1}$$

All online decisions obey the forecast-time information boundary

$$\mathcal{M}_t = \{i : \text{truth_available_at}(i) \le t\}, \tag{2}$$

so realized residuals and outcome statistics are used only after their corresponding ground truth becomes available. The prolem and proposed framework are summarized in Figure 1.

3.2 LAYER 1: BASE FORECAST AND STRUCTURED EVIDENCE

A series-specific Ridge model produces the Base center $\mu_{v,t}^0$. For mature historical forecast $i \in \mathcal{M}_t$, the rolling-origin residual is

$$e_{v,i}^0 = y_{v,i+\Delta} - \mu_{v,i}^0.$$

A Residual-Scale Ridge uses recent mature $h_{v,i} = \log(|e_{v,i}^0| + \epsilon)$ values to predict $s_{v,t}^0 = \exp(\hat{h}_{v,t})$. Historical residuals are standardized as

$$u_{v,i}^0 = \frac{e_{v,i}^0}{s_{v,i}^0 + \epsilon}.$$

A rolling pool first matches the target series, horizon, and time-of-day group and falls back to less specific but compatible groups when support is insufficient. For nominal coverage $1-\alpha$,

$$\boxed{L_{v,t}^0 = \mu_{v,t}^0 + s_{v,t}^0 q_t^-, \qquad U_{v,t}^0 = \mu_{v,t}^0 + s_{v,t}^0 q_t^+,} \quad q_t^{\pm} = Q_{\{\alpha/2,1-\alpha/2\}}(\mathcal{U}_t^W). \tag{3}$$

Both tails are estimated from the same signed standardized-residual distribution. The resulting empirical interval is a rolling probabilistic baseline rather than a conformal guarantee.

To characterize structural conditions potentially missing from the Base calibration, we construct a dynamic PTE graph from forecast-time data. Candidate paths are bounded, deduplicated, and ranked using normalized statistical dependency strength together with mature residual states aligned to the evidence time. For representative path P_j,

$$A_j = \left[\prod_{u \in P_j} (a_{u,j} + \epsilon)\right]^{1/|P_j|}, \quad C_j^{\text{sign}} = \left|\frac{\sum_u a_{u,j} \operatorname{sign}(z_{u,j}^0)}{\sum_u a_{u,j} + \epsilon}\right|, \tag{4}$$

where $a_{u,j} = 1 - \exp(-|z_{u,j}^0|)$ and $z_{u,j}^0$ is the mature standardized Base residual aligned with the relevant evidence time. Path relevance is

$$R_j = S_j^{\text{PTE}} A_j \frac{1 + C_j^{\text{sign}}}{2}, \qquad S_j^{\text{PTE}} = \min_{\ell \in P_j} \widetilde{\text{PTE}}_\ell. \tag{5}$$

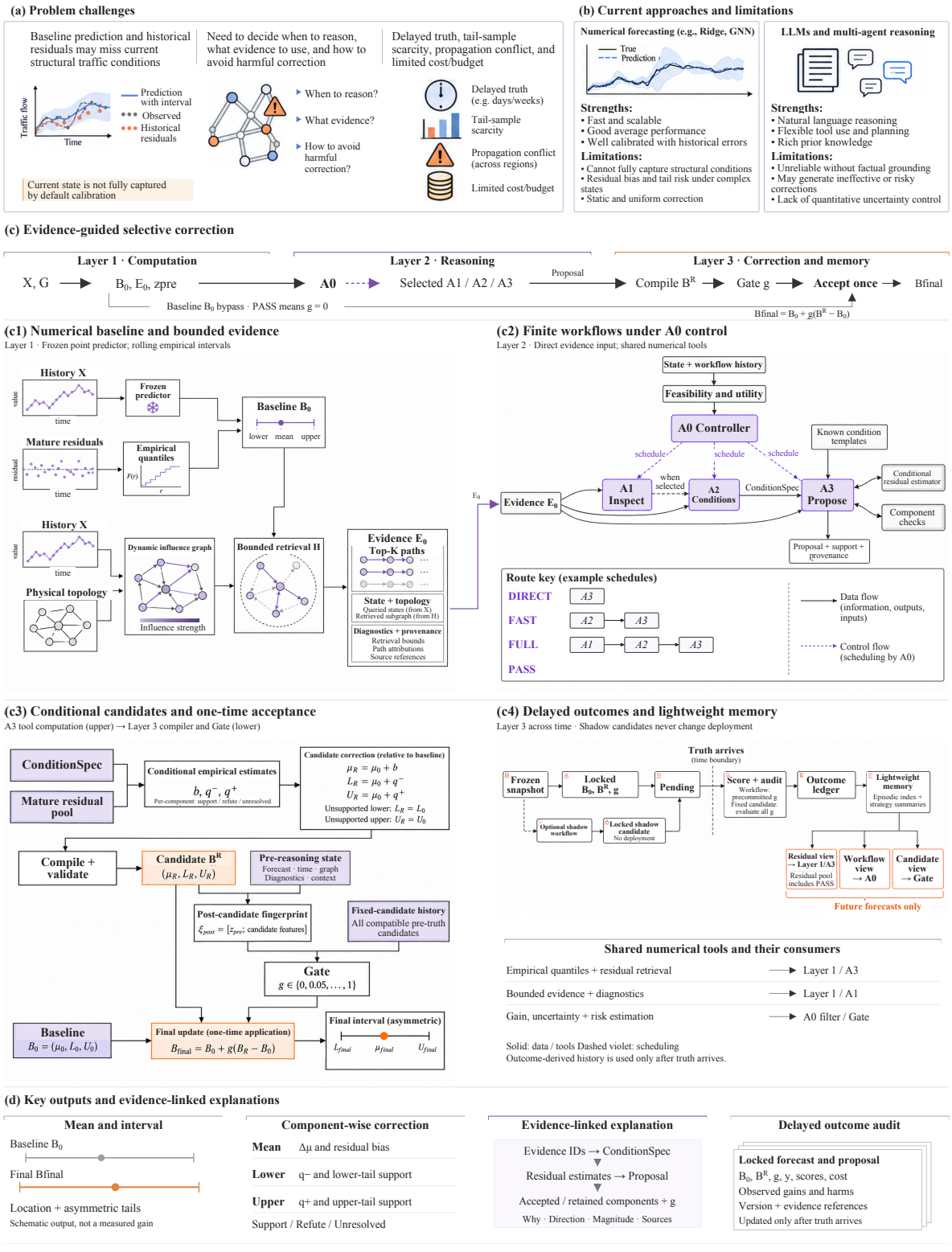


Figure 1: Overview of the proposed evidence-guided selective probabilistic correction framework.

C_j^{sign} reflects agreement among aligned residual states and does not represent the causal sign of PTE propagation. The selected components form primary evidence E_t^0, while other legal components remain in Reserve_t. Evidence scores are used only for retrieval and prioritization, not as correction magnitudes.

The Base prediction, recent temporal dynamics, graph statistics, diagnostics, and context are summarized by

$$\mathbf{z}_t^{\text{pre}} = [\mathbf{z}_t^{\text{forecast}}; \mathbf{z}_t^{\text{temporal}}; \mathbf{z}_t^{\text{graph}}; \mathbf{z}_t^{\text{diagnostic}}; \mathbf{z}_t^{\text{context}}].$$

3.3 Layer 2: Selective Reasoning and Conditional Correction

A0 is the Reasoning Controller. It selects among PASS, DIRECT, FAST, and FULL. The latter three respectively execute A3, A2→A3, and A1→A2→A3. Workflow selection is constrained by mature historical support, conservative utility, harmful-outcome risk, and execution cost:

$$\mathcal{W}_t^{\text{feas}} = \{\text{PASS}\} \cup \left\{ w : \begin{array}{l} n_{\text{eff},t}(w) \geq n_{\min}, \\ \underline{U}_t(w) - \lambda_c c_t(w) > 0, \\ \overline{R}_t^{\text{route}}(w) \leq \tau_{\text{route}}, \\ c_t(w) \leq c_{\max} \end{array} \right\}. \tag{6}$$

A0 selects only within $\mathcal{W}_t^{\text{feas}}$.

Under FULL, A1 checks temporal applicability, dependency strength, redundancy, coverage, and residual conflicts and may revise the evidence to E_t^*. A2 then converts the evidence used by the selected workflow into a structured ConditionSpec $\mathcal{C}_t$, specifying the affected forecast components, bounded condition fields, necessary conditions, support and counterevidence IDs, and unresolved fields. The agents therefore specify *which residual condition should be examined*, rather than directly generating numerical correction magnitudes.

A3 estimates the correction from the mature Base-residual view

$$\mathcal{D}_t^{\text{res}} = \{i \in \mathcal{M}_t : i \text{ is compatible in target, horizon, and model version}\}.$$

This set includes PASS cases and is deliberately independent of workflow selection. Let $\mathcal{H}_t(\mathcal{C}_t) \subseteq \mathcal{D}_t^{\text{res}}$ denote the condition-matched records with weights ω_{ti}. Their effective support is

$$n_{\text{eff},t} = \frac{(\sum_i \omega_{ti})^2}{\sum_i \omega_{ti}^2}.$$

The weighted local distribution is

$$\widehat{F}_t^{\text{loc}}(u) = \frac{\sum_i \omega_{ti} \mathbf{1}[u_i^0 \leq u]}{\sum_i \omega_{ti}}, \tag{7}$$

and limited local support is handled by hierarchical shrinkage,

$$\widehat{F}_t^{\text{cond}} = \gamma_t \widehat{F}_t^{\text{loc}} + (1 - \gamma_t)\widehat{F}_t^{\text{par}}, \qquad \gamma_t = \frac{n_{\text{eff},t}}{n_{\text{eff},t} + \kappa}. \tag{8}$$

The parent distribution relaxes the least-specific fields in $\mathcal{C}_t$ while retaining target/horizon and version compatibility.

The same conditional residual distribution determines both center and tail corrections:

$$\bar{u}_t^C = \mathbb{E}_{\widehat{F}_t^{\text{cond}}}[u], \qquad q_t^{C,-} = Q_{\alpha/2}(\widehat{F}_t^{\text{cond}}), \qquad q_t^{C,+} = Q_{1-\alpha/2}(\widehat{F}_t^{\text{cond}}). \tag{9}$$

Let $\delta_t^\mu = (s_t^0 + \epsilon)\bar{u}_t^C$ and $\delta_t^\pm = (s_t^0 + \epsilon)q_t^{C,\pm}$. After component-wise support and counterevidence checks, $m_t^\mu, m_t^-, m_t^+ \in \{0, 1\}$ determine the fixed candidate

$$\boxed{\begin{array}{l} \mu_t^R = \mu_t^0 + m_t^\mu \delta_t^\mu, \\ L_t^R = L_t^0 + m_t^- \left[(\mu_t^0 + \delta_t^-) - L_t^0\right], \\ U_t^R = U_t^0 + m_t^+ \left[(\mu_t^0 + \delta_t^+) - U_t^0\right]. \end{array}} \tag{10}$$

Hence an unsupported endpoint remains exactly at its Base value even when the center changes. DIRECT reuses a validated ConditionSpec but re-estimates its numerical magnitude from current mature history rather than copying a single historical correction. A3 returns B_t^R together with the ConditionSpec, support status, residual statistics, and evidence provenance.

3.4 Layer 3: Outcome-Grounded Deployment and Feedback

Layer 3 decides only how much of B_t^R to deploy. After candidate validity checks, define

$$\mathbf{d}_t^R = \frac{1}{s_t^0 + \epsilon}[\mu_t^R - \mu_t^0,\, L_t^R - L_t^0,\, U_t^R - U_t^0]^\top,$$

and $\boldsymbol{\xi}_t^{\text{post}} = [\mathbf{z}_t^{\text{pre}}; \mathbf{d}_t^R]$. The Gate retrieves comparable records from $\mathcal{D}_t^{\text{gate}} \subseteq \mathcal{M}_t$, which contains only candidates frozen before their ground truths became available.

For historical candidate i,

$$B_i(g) = B_i^0 + g(B_i^R - B_i^0), \qquad g \in \mathcal{G} = \{0, 0.05, \ldots, 1\}.$$

We evaluate center and interval quality using the unified score

$$S_i(B, y_i) = \lambda_\mu \left(\frac{y_i - \mu(B)}{s_i^0 + \epsilon} \right)^2 + \lambda_I \frac{IS_\alpha(B, y_i)}{s_i^0 + \epsilon}, \tag{11}$$

where

$$IS_\alpha(L, U; y) = U - L + \frac{2}{\alpha}(L - y)\mathbf{1}[y < L] + \frac{2}{\alpha}(y - U)\mathbf{1}[y > U].$$

The gain of deployment strength g is

$$G_i(g) = S_i(B_i^0, y_i) - S_i(B_i(g), y_i).$$

Using forecast-time similarity weights ν_{ti}, we estimate

$$\widehat{G}_t(g) = \frac{\sum_i \nu_{ti} G_i(g)}{\sum_i \nu_{ti}}, \qquad \underline{G}_t(g) = \widehat{G}_t(g) - \beta \widehat{\mathrm{SE}}_t(g),$$

where uncertainty is estimated using temporally or event-blocked historical samples. Historical cases with $G_i(g) < -\epsilon_{\text{harm}}$ provide the local harmful-correction risk estimate $\overline{R}_t(g)$.

Deployment is

$$\boxed{g_t = \arg\max_{g \in \mathcal{G}} [\underline{G}_t(g) - \eta g^2] \quad \text{s.t.} \quad \overline{R}_t(g) \le \tau_r, \qquad B_t^{\text{final}} = B_t^0 + g_t(B_t^R - B_t^0).} \tag{12}$$

Insufficient effective support or an invalid candidate yields $g_t = 0$. This constitutes an empirically conservative acceptance rule rather than a per-instance or coverage guarantee.

Before $y_{t+\Delta}$ becomes available, the state, workflow, ConditionSpec, Base forecast, fixed candidate, committed g_t, final forecast, provenance, cost, and versions remain in a Pending record. Once the outcome becomes available, the record is moved to the Outcome Ledger. The same mature Ledger provides three deliberately distinct historical views:

$$\mathcal{D}_t^{\text{res}} \to \text{A3}, \qquad \mathcal{D}_t^{\text{route}} \to \text{A0}, \qquad \mathcal{D}_t^{\text{gate}} \to \text{Gate}. \tag{13}$$

Residual estimation therefore uses all compatible Base records, workflow allocation uses only executed or shadow-executed workflow outcomes, and deployment uses only pre-truth frozen candidates.

For workflow w_i, routing utility is evaluated under the deployment strength committed before observing its outcome,

$$U_i^{\text{route}} = S_i(B_i^0, y_i) - S_i(B_i(g_i^{\text{pre}}), y_i),$$

preventing retrospective Gate optimization from inflating workflow utility. For cold-start states, low-budget Shadow execution freezes candidates without changing the published forecast. Early Shadow outcomes initialize Gate history; once Gate support becomes available, subsequent Shadow cases commit g_i^{pre} before outcome observation and can therefore provide valid workflow-utility records.

4 Experiments

We evaluate the framework through three questions corresponding to evidence conditioning, reasoning allocation, and candidate deployment: **RQ1:** How does evidence-conditioned residual correction affect point and interval forecasts across datasets and traffic states? **RQ2:** What accuracy–reasoning-cost trade-off results from selecting workflow depth instead of always using FULL reasoning? **RQ3:** How do candidate utility and realized correction outcomes vary with outcome-grounded deployment strength?

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate on XuanCheng (Wang et al., 2023), a 5-minute city-scale traffic-flow dataset with 124 intersections, and PEMS-BAY (Li et al., 2018), a 5-minute highway speed dataset with 325 sensors. We fix 10 target series per dataset across different spatial locations and network roles. Data splits and target locations are provided in Appendix B.1.

Baselines. The Base combines a series-specific Ridge forecaster, Residual-Scale Ridge, and rolling asymmetric empirical prediction intervals. We compare with six representative methods: the graph-based STGCN (Yu et al., 2018); the LLM-based TIME-LLM (Jin et al., 2024) and ST-LLM (Liu et al., 2024); and the probabilistic ASTGCN-DR (Zheng et al., 2025), DiffSTG (Wen et al., 2023), and SpecSTG (Lin et al., 2025).

Evaluation Metrics. The main comparisons report MAE, RMSE, PICP_{90}, and MPIW. Daily analyses and deployment diagnostics additionally use the interval score (IS), which combines interval width and miscoverage penalties. Reasoning usage is measured by live LLM calls per case. Coverage is assessed relative to the nominal 90% level, with interval width interpreted jointly with coverage. Definitions are provided in Appendix B.2.

Implementations. All hyperparameters are selected on Train/Validation data and frozen before Test. Forecasts and deployment decisions are fixed before their outcomes become available, and matured outcomes update only predefined online statistics. All LLM modules use gpt-4o-mini with a fixed configuration. Details are provided in Appendix B.3.

4.2 RQ1: EVIDENCE-CONDITIONED PROBABILISTIC CORRECTION

Main Results. Table 1 compares the resulting forecasts with numerical, LLM-based, and probabilistic forecasting methods. Our method achieves MAE/RMSE of 8.47/12.19 on XuanCheng and 1.13/1.84 on PEMS-BAY. The fine-tuned ST-LLM has lower point errors on both datasets. Compared with ASTGCN-DR, our method produces narrower intervals, with coverage closer to 90% on XuanCheng and nearly equal coverage on PEMS-BAY. DiffSTG and SpecSTG have lower coverage on both datasets. Their narrower intervals on XuanCheng come with more frequent miscoverage. The direct effect of augmenting the Base is examined separately in Table 2.

Table 1: Forecasting results on the fixed target series of XuanCheng and PEMS-BAY.

	XuanCheng				PEMS-BAY			
Method	MAE	RMSE	$\text{PICP}_{90}(\%)$	MPIW	MAE	RMSE	$\text{PICP}_{90}(\%)$	MPIW
STGCN	8.23	11.67	–	–	1.29	2.13	–	–
TIME-LLM	9.04	13.05	–	–	1.18	2.00	–	–
ST-LLM	7.71	10.85	–	–	1.09	1.82	–	–
ASTGCN-DR	8.53	11.74	86.99	35.15	1.25	1.99	89.43	5.26
DiffSTG	9.74	14.45	75.98	25.88	1.30	2.18	76.83	3.68
SpecSTG	9.65	13.60	73.18	24.99	1.80	3.18	81.46	5.14
Ours	8.47	12.19	88.38	33.66	1.13	1.84	89.44	4.78

Evidence Conditioning. In Table 2, removing evidence conditioning increases XuanCheng MAE from 8.473 to 8.809 and RMSE from 12.188 to 12.620, returning point errors to the Base level. On PEMS-BAY, MAE changes only from 1.133 to 1.134, retaining most of the gain over the Base. Its observed contribution is larger on XuanCheng. The full method reduces point errors and narrows intervals relative to the Base, while coverage changes from 89.317% to 88.376% on XuanCheng and from 89.723% to 89.436% on PEMS-BAY. We therefore examine IS over time to assess the balance between interval width and miscoverage.

Daily Variation. Figure 2 reports relative MAE and IS improvements over the Base. MAE improves on all ten test days for both datasets, whereas IS improves on seven days for XuanCheng and nine for PEMS-BAY. Point-error gains extend across the reported test days, while the width–miscoverage balance varies over time.

Table 2: Effects of evidence conditioning, workflow allocation, and the Gate.

	XuanCheng				PEMS-BAY			
Method	MAE↓	RMSE↓	PICP$_{90}$ (%)	MPIW	MAE↓	RMSE↓	PICP$_{90}$ (%)	MPIW
Base	8.809	12.619	89.317	34.692	1.148	1.867	89.723	4.857
w/o Evidence Cond.	8.809	12.620	89.149	34.537	1.134	1.844	89.455	4.781
Always FULL	8.216	11.742	88.099	33.185	1.125	1.832	89.545	4.784
w/o Gate	8.400	12.076	87.594	33.196	1.133	1.839	89.307	4.797
Ours	8.473	12.188	88.376	33.658	1.133	1.843	89.436	4.781

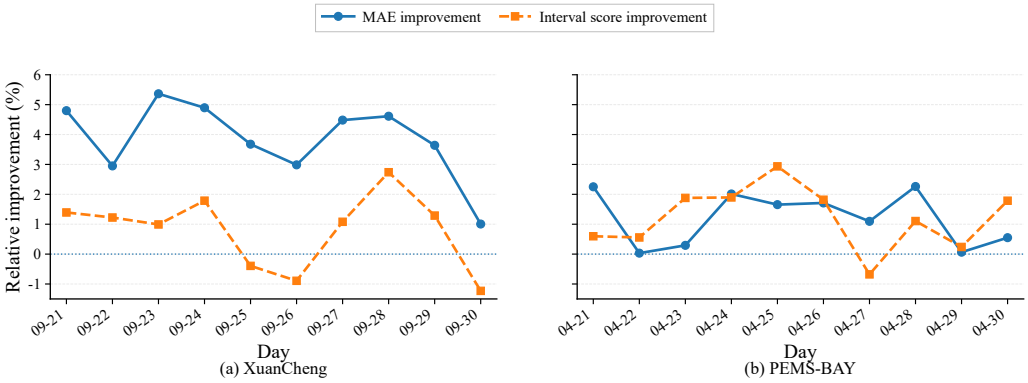


Figure 2: Daily relative improvements over the Base.

Speed-Regime Analysis. On PEMS-BAY, congestion-formation cases have larger center adjustments and MAE gains than stable free flow (Table 3). The lower-bound change is negative during formation and positive during free flow, indicating state-dependent correction. Appendix C.1 separates candidate gains from deployment strength in two tail-oriented cases.

4.3 RQ2: Selective Reasoning Allocation

Selective Reasoning. The controller selects FULL for a minority of forecast cases. On XuanCheng, 1,650 of 10,100 cases use FULL and 4,259 use PASS. On PEMS-BAY, 1,578 cases use FULL and 5,419 use PASS. FULL therefore accounts for 16.3% and 15.6% of cases, respectively. The route distributions in Appendix C.2 describe how computation is allocated. Table 4 reports the associated forecast errors and live-call usage.

Accuracy–Reasoning-Cost Trade-off. Compared with Always FULL, selective reasoning reduces live calls per case from 1.669 to 1.023 on XuanCheng and from 1.995 to 0.794 on PEMS-BAY, reductions of 38.7% and 60.2%. MAE increases from 8.216 to 8.473 and from 1.125 to 1.133, respectively. Selective reasoning retains lower MAE than the Base, with a smaller accuracy gap to Always FULL on PEMS-BAY.

4.4 RQ3: Outcome-Grounded Deployment

Selective Deployment. Removing the Gate lowers RMSE on both datasets and MAE on XuanCheng. PEMS-BAY MAE is unchanged at the reported precision (Table 2). The Gate raises interval coverage relative to this variant, although coverage remains below the Base. Its effect is therefore a change in the point-error and interval-coverage trade-off.

The PEMS-BAY diagnostics group fixed candidates by the Gate decision made before their outcomes became available (Appendix C.3). Mean candidate IS gain over the Base is -0.137 for suppressed candidates, 0.242 for partially deployed candidates, and 0.558 for fully deployed candidates. This ordering associates greater deployment strength with larger subsequent candidate gains.

Table 3: Correction across traffic regimes.

Traffic state	$\lvert\Delta\mu\rvert$	ΔL	MAE gain
Free flow	0.302	$+0.052$	0.019
Congestion	0.497	-0.053	0.090

Table 4: Live LLM calls per case and MAE.

	XuanCheng		PEMS-BAY	
Method	Calls/case↓	MAE ↓	Calls/case↓	MAE ↓
Always FULL	1.669	8.216	1.995	1.125
Ours	**1.023**	8.473	**0.794**	1.133

The appendix separately reports loss attenuation among candidates that would be harmful under full deployment.

Case Study. Figure 3 follows a XuanCheng forecast at node 4920 from candidate construction to outcome evaluation. The historical Gate selects $g = 0.80$ before the outcome is observed. Once the outcome becomes available, the Base interval misses it, whereas the candidate and deployed intervals cover it. Deployment narrows the candidate interval. The matured record then becomes available to the residual, routing, and deployment views.



Figure 3: Case study of evidence-conditioned correction and outcome-grounded deployment.

5 Conclusion

We propose **evidence-guided selective probabilistic reasoning**, which connects structural evidence to conditional residual estimation and outcome-grounded deployment. Agents identify relevant residual conditions, deterministic estimators compute center and tail corrections, and separate decisions control reasoning effort and correction strength. Experiments on XuanCheng and PEMS-BAY show lower point errors than the Base, dataset-dependent effects of evidence conditioning, and fewer live LLM calls than Always FULL at higher MAE. Candidate-level diagnostics further relate deployment strength to subsequent utility and quantify loss attenuation within harmful forced corrections.

AI USE STATEMENT

Generative AI tools were used solely in an assistive capacity for literature search and summarization, translation and language refinement, terminology consistency checks, and feedback on the clarity and presentation of the methodology and experimental design. They were also used to support the organization and editing of the manuscript. All AI-assisted material was critically reviewed and, where necessary, revised by the authors. Technical claims, experimental results, and citations were independently verified against the implementation, experimental records, and original sources. The authors made all research decisions and take full responsibility for the analyses, conclusions, and final content of this work.

REPRODUCIBILITY STATEMENT

We support reproducibility through explicit specification of the model, evaluation protocol, and experimental configuration. Section 3 describes the complete framework and forecast-time information boundary. Appendix A provides additional details on evidence construction, conditional residual estimation, candidate validation, historical utility and risk estimation, ledger views, and chronological online execution. Appendix B reports the data splits, fixed target series, implementation settings, and evaluation protocol. All hyperparameters are selected using Train/Validation data and frozen before Test. An anonymized code package containing the implementation, configuration files, and scripts for reproducing the experiments is provided as supplementary material.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=gerNCVqqtR.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 11733–11763, 2024. URL https://proceedings.mlr.press/v235/du24e.html.

Isaac Gibbs and Emmanuel Candès. Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://papers.neurips.cc/paper_files/paper/2021/hash/0d441de75945e5acbc865406fc9a2559-Abstract.html.

Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. MetaGPT: Meta programming for a multi-agent collaborative framework. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/6507b115562bb0a305f1958ccc87355a-Abstract-Conference.html.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.01728.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

Lequan Lin, Dai Shi, Andi Han, and Junbin Gao. SpecSTG: A fast spectral diffusion framework for probabilistic spatio-temporal traffic forecasting. In *ICLR 2025 Workshop on Deep Learning for Time Series*, 2025. URL https://openreview.net/forum?id=qBuCq3ER02.

Chenxi Liu, Sun Yang, Qianxiong Xu, Zhishuai Li, Cheng Long, Ziyue Li, and Rui Zhao. Spatial-temporal large language model for traffic prediction. In *2024 25th IEEE International Conference on Mobile Data Management (MDM)*, pp. 31–40. IEEE, 2024. doi: 10.1109/MDM61037.2024.00025.

Yimin Wang, Yixian Chen, Guilong Li, Yuhuan Lu, Zhaocheng He, Zhi Yu, and Weiwei Sun. City-scale holographic traffic flow data based on vehicular trajectory resampling. *Scientific Data*, 10 (1):57, 2023. doi: 10.1038/s41597-022-01850-0.

Haomin Wen, Youfang Lin, Yutong Xia, Huaiyu Wan, Qingsong Wen, Roger Zimmermann, and Yuxuan Liang. DiffSTG: Probabilistic spatio-temporal graph forecasting with denoising diffusion models. In *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, pp. 1–12. Association for Computing Machinery, 2023. doi: 10.1145/3589132.3625614.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversations. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=BAakY1hNKS.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 753–763, 2020. doi: 10.1145/3394486.3403118.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 3634–3640, 2018. doi: 10.24963/ijcai.2018/505.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317. URL https://ojs.aaai.org/index.php/AAAI/article/view/26317.

Vincent Zhihao Zheng, Seongjin Choi, and Lijun Sun. Probabilistic traffic forecasting with dynamic regression. *Transportation Science*, 59(4):689–707, 2025. doi: 10.1287/trsc.2024.0560.

Yitong Zhou, Yucong Luo, Mingyue Cheng, Qi Liu, Jiahao Wang, Daoyu Wang, and Enhong Chen. Time series forecasting via reasoning: A slow-thinking approach with reinforcement fine-tuned LLMs, 2026. URL https://arxiv.org/abs/2506.10630v4. Version 4, revised August 24, 2026.

A Supplementary Method Details

This appendix complements the methodology in the main paper with details on evidence construction, conditional retrieval, outcome-grounded routing, and the chronological feedback protocol.

A.1 Evidence Construction Details

The dynamic PTE graph is estimated only from observations available at the current forecast origin. The physical graph provides candidate structural context, whereas PTE represents directed statistical prediction dependency; neither is interpreted as a directly identified causal response.

For node u on representative path P_j, let $\tau_{u,j}$ be the mature historical forecast origin aligned with its evidence time:

$$z^0_{u,j} = \frac{e^0_{u,\tau_{u,j}}}{s^0_{u,\tau_{u,j}} + \epsilon}, \qquad a_{u,j} = 1 - \exp(-|z^0_{u,j}|). \tag{14}$$

The path-level residual activation, residual-sign consistency, and normalized dependency bottleneck are

$$\begin{aligned} A_j &= \left[\prod_{u \in P_j} (a_{u,j} + \epsilon) \right]^{1/|P_j|}, \\ C^{\text{sign}}_j &= \left| \frac{\sum_{u \in P_j} a_{u,j} \operatorname{sign}(z^0_{u,j})}{\sum_{u \in P_j} a_{u,j} + \epsilon} \right|, \\ S^{\text{PTE}}_j &= \min_{\ell \in P_j} \widetilde{\text{PTE}}_\ell. \end{aligned} \tag{15}$$

The ranking score is

$$R_j = S^{\text{PTE}}_j A_j \frac{1 + C^{\text{sign}}_j}{2}. \tag{16}$$

Residual-sign consistency reflects agreement among historical residual states, not the causal sign of PTE propagation. Similarly, R_j is used only for evidence selection and cannot directly generate a forecast correction.

Primary components form E^0_t and other legal active components remain in Reserve_t. Each retained component stores its relevant time, dependency statistics, residual state, structural context, and provenance.

A.2 Conditional Retrieval and Hierarchical Estimation

The mature Base-residual view is

$$\mathcal{D}^{\text{res}}_t = \{ i \in \mathcal{M}_t : i \text{ is compatible in target, horizon, and model version} \}. \tag{17}$$

It contains all compatible mature Base predictions, including PASS cases. Given ConditionSpec $\mathcal{C}_t$, A3 retrieves $\mathcal{H}_t(\mathcal{C}_t) \subseteq \mathcal{D}^{\text{res}}_t$.

Historical matching is performed using forecast-time states only. A generic weight is

$$\omega_{ti} = \chi_{ti} \exp\left(-\frac{d(\mathbf{z}_t, \mathbf{z}_i)}{\tau_s} \right) \exp\left(-\frac{t - t_i}{\tau_{\text{age}}} \right) q^{\text{data}}_i, \tag{18}$$

where χ_{ti} is a hard compatibility indicator. The corresponding effective support is

$$n_{\text{eff},t} = \frac{(\sum_i \omega_{ti})^2}{\sum_i \omega^2_{ti}}. \tag{19}$$

The weighted local empirical distribution is

$$\widehat{F}^{\text{loc}}_t(u) = \frac{\sum_i \omega_{ti} \mathbf{1}[u^0_i \le u]}{\sum_i \omega_{ti}}. \tag{20}$$

When local support is insufficient,

$$\widehat{F}^{\text{cond}}_t = \gamma_t \widehat{F}^{\text{loc}}_t + (1 - \gamma_t) \widehat{F}^{\text{par}}_t, \qquad \gamma_t = \frac{n_{\text{eff},t}}{n_{\text{eff},t} + \kappa}. \tag{21}$$

The parent distribution relaxes the least-specific ConditionSpec fields while preserving target/horizon and model-version compatibility.

The shrinkage coefficient above addresses statistical uncertainty in estimating the residual distribution. It is conceptually different from the Layer 3 deployment strength g_t, which controls how much of a fixed candidate is finally used.

A.3 Component-Wise Candidate Validation

A3 separately evaluates evidence for the predictive center, lower endpoint, and upper endpoint. The result for each component may be interpreted as supported, contradicted, or unresolved; only supported corrections are activated in the candidate. This allows, for example, a center correction with unchanged endpoints or a one-sided tail correction with an unchanged center.

Before Layer 3, the candidate must satisfy:

- finite numerical values and ordered interval endpoints;
- target, horizon, unit, and version compatibility;
- valid evidence and ConditionSpec provenance;
- predefined correction-magnitude constraints; and
- strict forecast-time information eligibility.

Candidates failing these checks receive $g_t = 0$.

A.4 Historical Utility and Risk Estimation

For a fixed historical candidate,

$$B_i(g) = B_i^0 + g(B_i^R - B_i^0), \qquad g \in \{0, 0.05, \ldots, 1\}. \tag{22}$$

Its gain is

$$G_i(g) = S_i(B_i^0, y_i) - S_i(B_i(g), y_i), \tag{23}$$

using the unified center-plus-interval score defined in the main paper.

For locally comparable cases with weights ν_{ti},

$$\widehat{G}_t(g) = \frac{\sum_i \nu_{ti} G_i(g)}{\sum_i \nu_{ti}}, \qquad n_{\mathrm{eff},t}^{\mathrm{gate}} = \frac{(\sum_i \nu_{ti})^2}{\sum_i \nu_{ti}^2}. \tag{24}$$

The conservative deployment criterion uses

$$\underline{G}_t(g) = \widehat{G}_t(g) - \beta \widehat{\mathrm{SE}}_t(g), \tag{25}$$

where uncertainty is estimated using temporal or event blocks to avoid interpreting highly correlated cases as independent evidence.

A harmful correction satisfies

$$G_i(g) < -\epsilon_{\mathrm{harm}}. \tag{26}$$

A conservative local risk estimate $\overline{R}_t(g)$ is obtained from mature comparable cases. Insufficient case or event support is treated as unknown risk and prevents nonzero deployment.

The same principle applies to workflow allocation. For workflow w_i, only the deployment strength g_i^{pre} committed before its outcome became available may be used to calculate

$$U_i^{\mathrm{route}} = S_i(B_i^0, y_i) - S_i(B_i(g_i^{\mathrm{pre}}), y_i). \tag{27}$$

The retrospectively best Gate is never substituted for g_i^{pre} when estimating workflow utility.

A.5 Ledger Views and Cold-Start Feedback

A record remains Pending until its target ground truth becomes available. Pending records store the forecast-time state, evidence, workflow, ConditionSpec, Base forecast, fixed candidate, committed deployment strength, published forecast, provenance, cost, and versions, but cannot contribute outcome statistics.

After maturation, the Outcome Ledger exposes three separate views:

$$\mathcal{D}_t^{\mathrm{res}} \to \mathrm{A3}, \qquad \mathcal{D}_t^{\mathrm{route}} \to \mathrm{A0}, \qquad \mathcal{D}_t^{\mathrm{gate}} \to \mathrm{Gate}. \tag{28}$$

This separation prevents workflow-selection bias from entering conditional residual estimation and prevents hindsight-optimized deployment from entering workflow selection.

Cold-start learning uses two Shadow stages. Initially, formal deployment remains at the Base forecast while a Shadow workflow may freeze a candidate before the outcome is visible. Once that outcome matures, the fixed candidate can be evaluated over the Gate grid, providing candidate-level history. After sufficient Gate support has accumulated, subsequent Shadow cases commit their Gate before observing the outcome; only these precommitted outcomes provide valid workflow-utility history.

A.6 Chronological Online Execution

At forecast origin t, the implementation proceeds in the following order:

1. mature newly available Pending records;
2. update eligible rolling residual statistics;
3. generate the Base forecast B_t^0;
4. construct the valid dynamic dependency graph and evidence packet;
5. construct $\mathbf{z}_t^{\mathrm{pre}}$;
6. estimate workflow support, utility, risk, and cost;
7. select PASS, DIRECT, FAST, or FULL through A0;
8. construct $\mathcal{C}_t$ and estimate the fixed candidate B_t^R;
9. validate the candidate;
10. retrieve compatible frozen historical candidates and select g_t;
11. publish $B_t^{\mathrm{final}} = B_t^0 + g_t(B_t^R - B_t^0)$;
12. optionally execute a budgeted Shadow workflow; and
13. freeze all current information in the Pending queue.

All model-selection thresholds, retrieval parameters, shrinkage coefficients, workflow constraints, score weights, risk thresholds, and Shadow budgets are selected before Test using the Train/Validation protocol.

B Experimental Details

B.1 Data Splits and Target Locations

XuanCheng (Wang et al., 2023) contains city-scale traffic flow observations at 5-minute intervals over 124 intersections, while PEMS-BAY (Li et al., 2018) contains 5-minute traffic speed observations from 325 highway sensors in the San Francisco Bay Area. For each dataset, we fix 10 target locations before formal evaluation and keep them unchanged across all experiments. Figure 4 shows the spatial distribution of the full network and the selected targets, while Table 5 summarizes the chronological data splits.

B.2 Evaluation Metrics

Let y_i and $\hat{y}_i$ denote the ground-truth value and point prediction for case i, respectively, and let $[L_i, U_i]$ denote its 90% prediction interval over N evaluation cases. We compute the mean absolute error (MAE) as

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i|\,, \tag{29}$$

and the root mean squared error (RMSE) as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}. \tag{30}$$

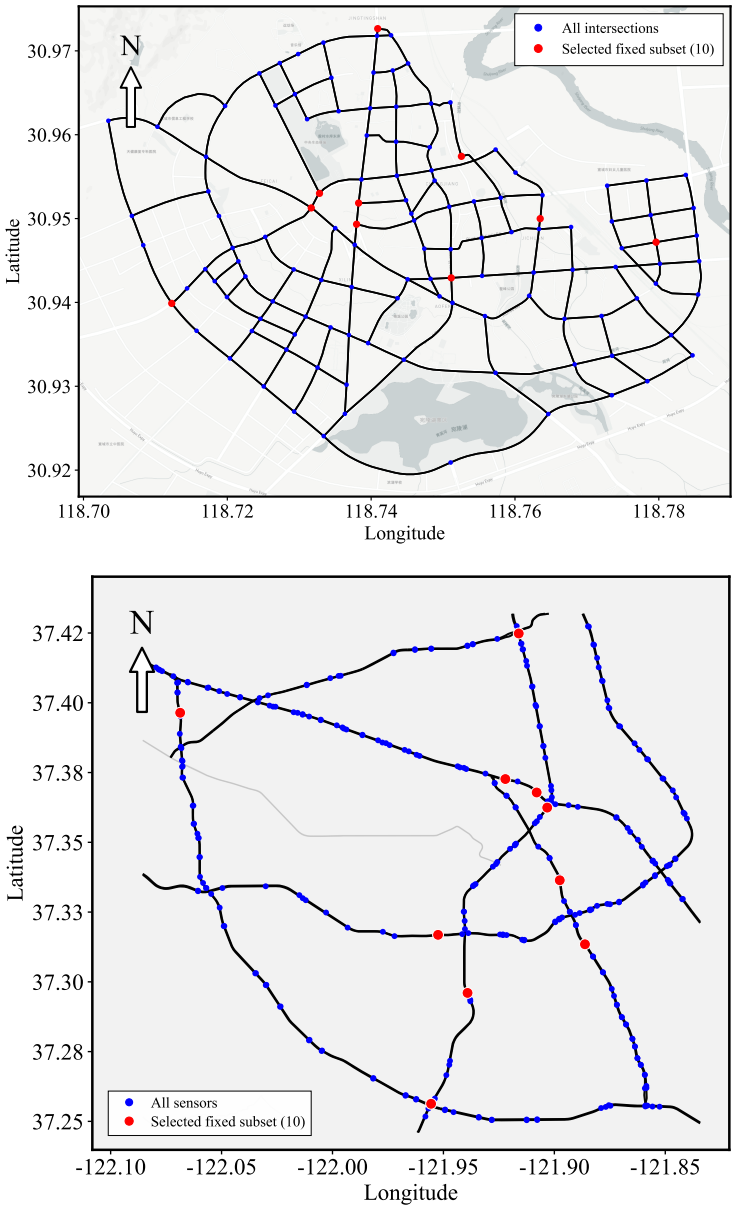


Figure 4: Fixed target locations in XuanCheng (up) and PEMS-BAY (down).

For interval evaluation, the empirical 90% prediction interval coverage probability is

$$\mathrm{PICP}_{90} = \frac{100}{N} \sum_{i=1}^{N} \mathbb{I}(L_i \le y_i \le U_i)\,, \tag{31}$$

where PICP_{90} is reported as a percentage. The mean prediction interval width (MPIW) is

$$\mathrm{MPIW} = \frac{1}{N} \sum_{i=1}^{N} (U_i - L_i)\,. \tag{32}$$

We further report the interval score (IS), which jointly accounts for interval width and miscoverage. For a 90% prediction interval, $\alpha = 0.1$, and

$$\mathrm{IS} = \frac{1}{N} \sum_{i=1}^{N} \left[(U_i - L_i) + \frac{2}{\alpha}(L_i - y_i)\mathbb{I}(y_i < L_i) + \frac{2}{\alpha}(y_i - U_i)\mathbb{I}(y_i > U_i) \right]. \tag{33}$$

Reasoning cost is measured by the average number of live LLM calls per case:

$$\text{Calls/Case} = \frac{1}{N}\sum_{i=1}^{N} c_i, \tag{34}$$

where c_i denotes the number of live LLM calls issued for case i.

For the daily analyses, we report the relative improvement of Ours over the Base for MAE and IS. For either metric M, we define

$$\text{RI}(M) = \frac{M_{\text{Base}} - M_{\text{Ours}}}{M_{\text{Base}}} \times 100\%. \tag{35}$$

Since lower values are preferred for both MAE and IS, a positive relative improvement indicates that Ours improves over the Base, whereas a negative value indicates degradation.

Lower MAE, RMSE, and IS indicate better performance. MPIW measures interval sharpness: a smaller width is favorable only in conjunction with adequate coverage. PICP_{90} is interpreted relative to the nominal 90% level. IS combines width and miscoverage penalties, while separate coverage and width summaries describe the underlying trade-off.

B.3 Implementation and Evaluation Protocol

Table 5 summarizes the chronological data splits. All hyperparameters for evidence retrieval, conditional residual estimation, reasoning allocation, and deployment are selected using only the Train/Validation data and frozen before Test. The same frozen configuration is used for both the main and ablation experiments.

Table 5: Chronological data splits.

Dataset	Train	Validation	Test	Resolution
XuanCheng	Sep. 2–14, 2020	Sep. 15–20, 2020	Sep. 21–30, 2020	5 min
PEMS-BAY	Apr. 3–14, 2017	Apr. 15–20, 2017	Apr. 21–30, 2017	5 min

Evaluation follows a strictly chronological online protocol. At each forecast origin, the prediction, reasoning workflow, candidate correction, and deployment decision are fixed before the corresponding outcome becomes available. Once an outcome matures, it may update only the predefined online residual, routing, and deployment statistics used by subsequent forecasts. Model parameters, retrieval settings, selection thresholds, and deployment rules remain fixed throughout Test. All LLM-based modules use gpt-4o-mini with a fixed configuration.

C Additional Experiment Results

The following analyses distinguish candidate-level corrections, workflow allocation, and realized deployment outcomes. Candidate gains evaluate the unattenuated forecast against the Base after the outcome is available. Deployment diagnostics assess how the selected strength changes the effect of that fixed candidate.

C.1 PEMS-BAY Speed-Regime Cases

Table 6 presents two PEMS-BAY cases with tail-oriented corrections. In both, the magnitude of the reported lower-bound change exceeds that of the center change. Their candidate IS gains differ: 15.411 for the formation case and 0.216 for the precursor case. These values describe the proposed candidate relative to the Base. The accompanying g values record how much of each candidate is selected for deployment. Candidate IS gain is not the score gain of the final forecast.

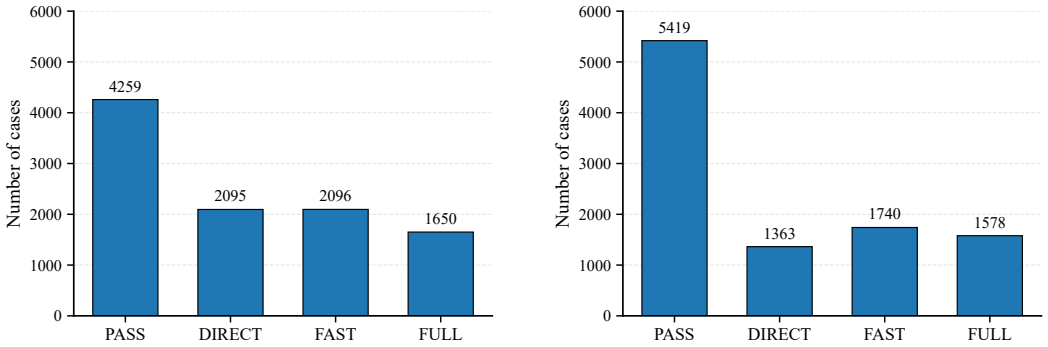
C.2 Reasoning Route Allocation

Figure 5 reports the selected workflow frequencies. PASS retains the Base without a formal correction workflow. DIRECT, FAST, and FULL invoke A3, A2 followed by A3, and A1 followed by A2

Table 6: PEMS-BAY speed-regime cases. IS gain compares the candidate with the Base; g is the selected deployment strength.

Case	Traffic state	g	$\lvert\Delta\mu\rvert$	ΔL	IS gain
A	Formation	0.55	0.164	-0.834	15.411
B	Precursor	0.95	0.219	-0.594	0.216

Case A: `target_404370_20170422_185500`.
Case B: `target_402365_20170424_204500`.
IS gain is measured from the Base forecast to the evidence-conditioned candidate.



(a) XuanCheng (b) PEMS-BAY

Figure 5: Routing decisions of the selective reasoning controller.

and A3, respectively. These counts describe the selected workflow, rather than a measured minimum reasoning depth required by each case. Their accuracy and live-call trade-off is reported in Table 4. PASS is also distinct from a Gate decision of $g = 0$, which suppresses a candidate after it has been constructed.

C.3 Gate Deployment Diagnostics

Table 7 separates candidate utility from the effect of deployment. In the upper panel, cases are grouped by the committed Gate decision. AE gain compares the Base center with the full candidate center, and IS gain compares the Base interval with the full candidate interval. Positive values indicate lower candidate loss. Thus, the nonzero gains in the $g = 0$ group describe suppressed candidates, not changes to the published forecast, which remains at the Base. Both AE and IS gains increase across the reported suppressed, partial, and full-deployment groups. These are separate diagnostics of the joint score used by the Gate in Eq. equation 11.

In the lower panel, 36.2% of attenuated cases would worsen IS under forced full deployment. The attenuation and avoidance rates refer to this harmful subset: the Gate reduces harm in 100.0% of these cases and fully avoids it relative to the Base in 52.0%. Mean IS harm decreases from 2.130 to 0.659, a 69.1% reduction. Returning to the Base with $g = 0$ necessarily avoids correction-induced loss. More generally, IS is convex in the interval endpoints, so interpolation toward the Base also attenuates loss. The lower panel therefore measures realized harm reduction within this subset. The upper panel separately describes the association between candidate utility and deployment strength.

Table 7: Gate diagnostics on PEMS-BAY. The upper panel reports full-candidate gains by committed deployment strength. The lower panel reports harm under forced deployment and its attenuation by the Gate.

Candidate utility by deployment			
Gate decision	Cases	AE gain	IS gain
Suppressed, $g = 0$	1,463	0.010	-0.137
Partial, $0 < g < 1$	2,491	0.035	0.242
Full, $g = 1$	521	0.114	0.558
Harm avoidance under attenuation			
Harmful under forced $g = 1$			36.2%
Harm attenuated by Gate			100.0%
Harm fully avoided			52.0%
Mean IS harm under forced deployment			2.130
Mean IS harm after gating			0.659