ADAPTIVE MULTI-RESOLUTION INFERENCE FOR FROZEN TIME-SERIES FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-series foundation models (TSFMs) have demonstrated zero-shot forecasting capabilities, but standard inference typically queries the model only once at the original temporal resolution. This may leave predictive information at other time scales underused. We study how multiple temporal resolutions can improve forecasts at the original resolution when the foundation model is fully frozen and the number of queries is limited. To this end, we propose Adaptive Multi-Resolution Inference (AMRI). AMRI aggregates the observed history to multiple temporal resolutions and queries the same frozen TSFM at each resolution. The outputs predict different temporal aggregates of the same future trajectory and therefore share known structural relationships. Using these aggregation relationships and forecast reliability estimated from historical errors before testing, AMRI reconciles the outputs through closed-form weighted least squares to obtain the final forecast at the original resolution. Adaptivity comes from reliability calibration for the target series. The resolution set and calibrated weights remain fixed during testing. The entire procedure requires no gradient-based training, parameter updates, or additional learned forecasting modules. Under unbiased forecast errors and exact population inverse-covariance weighting, we provide an expected squared-error guarantee for the reconciled forecast relative to the original-resolution forecast. Experiments across multiple frozen TSFMs and forecasting benchmarks demonstrate the effectiveness of AMRI. Code is available in the anonymous repository.

1 INTRODUCTION

Time-series foundation models (TSFMs) use large-scale pretraining to support forecasting on new time series. Models such as Chronos (Ansari et al., 2024), TimesFM (Das et al., 2024), and Moirai (Woo et al., 2024) demonstrate this zero-shot capability. A common practice is to query the model once using recent observations at their original sampling frequency. The same history can also be organized at different temporal resolutions. We ask: **with a frozen model, a limited number of queries, and a fixed input-length budget per query, how can forecasts at different temporal resolutions improve prediction at the original resolution?**

Changing temporal resolution is a simple way to construct model inputs. Original observations preserve fine-grained detail, while averaging consecutive observations reduces local fluctuations and emphasizes variation over longer time scales. Resolution also determines how much history a fixed number of input points can cover: 512 original observations span 512 time steps, whereas 512 averages of four-step blocks span 2048 time steps when enough history is available. Multi-resolution queries can thus change both the representation of history and its temporal coverage. We examine the benefit of combining multi-resolution forecasts over the same history, and the additional benefit of expanding that history.

Prior work exploits multiple time scales through multi-rate sampling and cross-scale modeling (Shabani et al., 2023; Wang et al., 2024). Adaptation of pretrained models includes parameter-efficient updates, test-time updates, and lightweight forecasting or correction components (Medeiros et al., 2025; Kim et al., 2025; Lee et al., 2025; Dai et al., 2026). SMR^2 (Gao et al., 2026b) and RGMR (Gao et al., 2026a) also combine multi-resolution queries to frozen models with fitted residual correctors. We keep the model's parameters and internal state fixed, adjusting the input resolution and

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

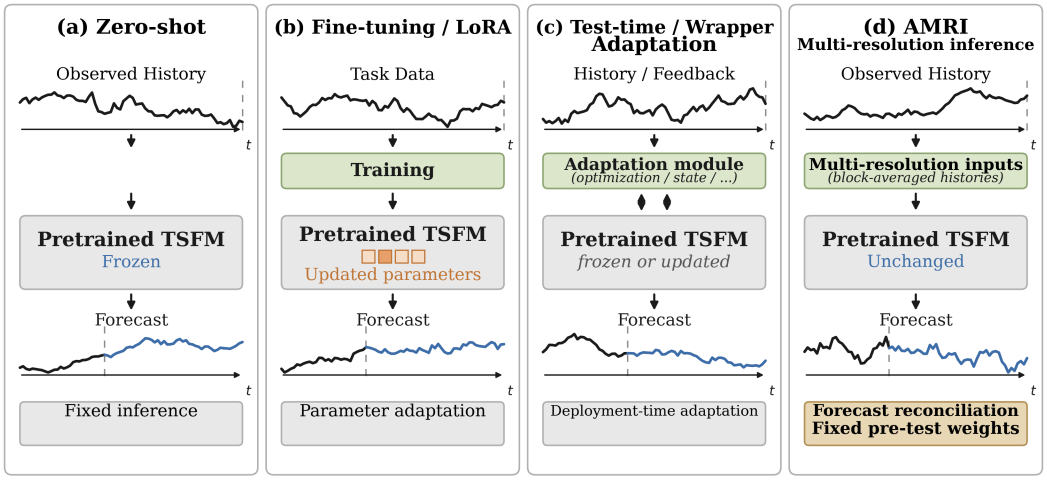

Figure 1: Where adaptation enters forecasting. (a) Direct zero-shot forecasting. (b) Fine-tuning or LoRA. (c) Test-time or wrapper adaptation. (d) AMRI queries the same frozen TSFM at multiple resolutions and combines forecasts using weights calibrated before testing. Curves are schematic. Dashed lines mark forecast origins.

forecast combination. Figure 1 contrasts this approach with direct zero-shot forecasting, fine-tuning, and test-time or wrapper adaptation.

Multi-resolution queries cover the same future interval but predict different targets. For example, four predictions at the original resolution target four future observations, while two predictions at a resolution formed by averaging pairs of time steps target two block means over that interval. Forecasting a block-averaged history can give different results from averaging forecasts made at the original resolution. The central combination problem is how to use these additional block-mean predictions to improve the prediction at each time step. Forecast reconciliation combines related forecasts into consistent predictions using known aggregation relationships (Hyndman et al., 2011; Wickramasuriya et al., 2019). Temporal reconciliation has also been applied to zero-shot TimeGPT forecasts (Calderón González & Gudiño-Ochoa, 2026). Building on this foundation, we study multi-resolution inference for frozen TSFMs under a specified query budget, with controls for input history and forecast reliability.

We propose **Adaptive Multi-Resolution Inference (AMRI)**. AMRI constructs block-averaged histories at a small set of predefined resolutions and queries the same frozen TSFM at each one. It then uses the known aggregation relationships and a closed-form weighted least-squares solution to combine these outputs into an original-resolution forecast. For each resolution and forecast position, AMRI estimates the forecast-error variance on a pre-test calibration segment, regularizes it, and uses its inverse as the weight. Here, adaptive refers to reliability calibration for the target series. The resolutions and calibrated weights remain fixed throughout testing, with no gradient-based training or additional forecasting model.

Our contributions can be summarized below:

1. **Multi-resolution inference for frozen TSFMs.** We formulate inference under query and input-length constraints and propose AMRI to reconcile a frozen model's multi-resolution forecasts using pre-test reliability calibration.
2. **Forecast disagreement and risk analysis.** We characterize how cross-resolution disagreement corrects forecasts and apply classical generalized least-squares theory to establish an expected squared-error guarantee under unbiased errors and exact population inverse-covariance weighting.
3. **Empirical evaluation.** Across seven frozen TSFMs, eight datasets, and four forecast horizons, controlled experiments assess multi-resolution forecast combination, history coverage, and reliability calibration.

2 Related Work

Time-series foundation models and adaptation. Time-series foundation models such as Chronos (Ansari et al., 2024), Chronos-2 (Ansari et al., 2025), TimesFM (Das et al., 2024), Moirai (Woo et al., 2024), Tiny Time Mixers (Ekambaram et al., 2024), MOMENT (Goswami et al., 2024), UniTS (Gao et al., 2024), and Lag-Llama (Rasul et al., 2024) enable transferable forecasting through large-scale pretraining. Downstream adaptation has been explored through reprogramming (Jin et al., 2024; Zhou et al., 2023), prompting (Cao et al., 2024; Xue & Salim, 2024), parameter-efficient updates (Medeiros et al., 2025), and test-time adaptation (Kim et al., 2025). Closer to our frozen setting, recent work perturbs model inputs (Hua et al., 2026) or fits lightweight components around a frozen forecaster to exploit deployment-time information: ELF (Lee et al., 2025) uses a lightweight forecaster and combiner, while ORCA (Dai et al., 2026) uses a black-box residual refiner. Our method queries the same frozen TSFM on block-averaged histories and combines the resulting forecasts using their known aggregation relationships.

Multi-scale time-series forecasting. Multi-scale structure has been incorporated through temporal aggregation (Athanasopoulos et al., 2017), multi-rate sampling (Shabani et al., 2023), frequency decomposition (Zhou et al., 2022), hierarchical representations (Challu et al., 2023), and cross-scale interaction in TimeMixer (Wang et al., 2024) and Pathformer (Chen et al., 2024). SMR^2 (Gao et al., 2026b) fits residual correctors to queries at several resolutions. RGMR (Residual-Guided Multi-Resolution Refinement) (Gao et al., 2026a) forms projected histories by block averaging and upsampling, then applies fitted residual predictors in a coarse-to-fine cascade. AMRI instead retains each output at its native resolution and reconciles these predictions as known aggregates of one fine-resolution future.

Forecast reconciliation. Forecast reconciliation adjusts a collection of related forecasts so that they become mutually consistent with known aggregation relationships. Unlike directly averaging forecasts of the same target, reconciliation accounts for the different aggregates being predicted. Classical optimal combination (Hyndman et al., 2011) uses forecast-error statistics to construct a coherent linear combination. Minimum-trace (MinT) reconciliation (Wickramasuriya et al., 2019) chooses covariance-based weights to minimize total reconciled error variance. Temporal reconciliation applies aggregation constraints across time scales (Athanasopoulos et al., 2017). Cross-temporal methods also incorporate aggregation across series (Kourentzes & Athanasopoulos, 2019; Di Fonzo & Girolimetto, 2023). Learning-based approaches estimate the mapping from base forecasts to coherent predictions (Spiliotis et al., 2021; Tsiourvas et al., 2024; Wang et al., 2025), or train forecasting models to produce coherent outputs directly (Rangapuram et al., 2021). Calderón González & Gudiño-Ochoa (2026) evaluate temporal reconciliation for photovoltaic forecasting, including zero-shot TimeGPT, using an hourly–daily–weekly hierarchy and weights re-estimated at each forecast cutoff. We use the established weighted-least-squares solution to study query budgets and input-history policies across seven frozen TSFMs, with calibration weights fixed before TEST.

3 Problem Setting

Let $\mathbf{y}_{1:t} = [y_1, \ldots, y_t]^\top$ denote the observed history of a target series at forecast origin t. Given a forecast horizon P, the task is to predict

$$\mathbf{y}_t^{\mathrm{fut}} = \mathbf{y}_{t+1:t+P} \in \mathbb{R}^P. \tag{1}$$

Let $f_{\mathrm{FM}}(\cdot, \cdot; h)$ denote the h-step point-forecast interface of a pretrained TSFM. The model may optionally receive d exogenous covariates $\mathbf{x}_{1:t} \in \mathbb{R}^{t \times d}$, temporally aligned with the target and available at the forecast origin. Let L be the per-query input-length budget. The single-resolution baseline uses the most recent L observations, or all available observations when $t < L$:

$$\hat{\mathbf{y}}_t^{(1)} = f_{\mathrm{FM}}(\mathbf{y}_{\max(1,t-L+1):t}, \mathbf{x}_{\max(1,t-L+1):t}; P) \in \mathbb{R}^P, \tag{2}$$

with the covariate argument omitted when unavailable or unsupported.

At forecast origin t, the available information $\mathcal{H}_t$ consists of the observed history $\mathbf{y}_{1:t}$, covariates $\mathbf{x}_{1:t}$, and errors of earlier forecasts whose targets have all been observed. A frozen multi-resolution procedure queries one pretrained model on deterministically aggregated histories, keeping its parameters and internal state fixed. All decisions use only $\mathcal{H}_t$.

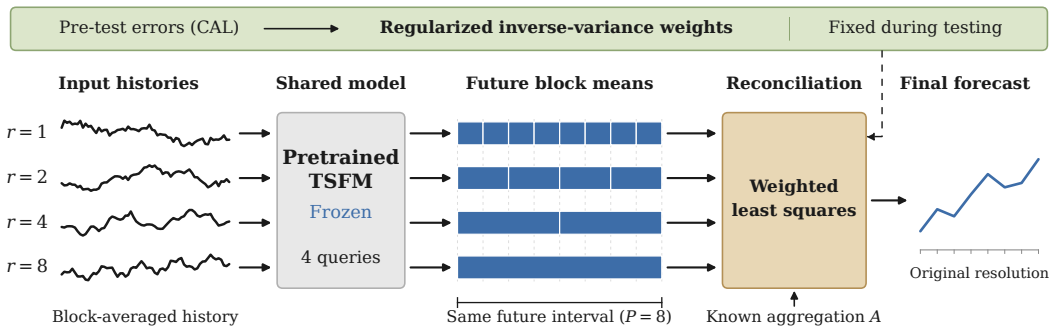

Figure 2: AMRI calibration and inference. Four queries to the same frozen TSFM predict 8, 4, 2, and 1 block means over the same future interval ($P = 8$). Weighted least squares combines them using the known aggregation matrix A and weights calibrated before testing. Curves are schematic.

The admissible resolutions are $\mathcal{A}_P = \{r \in \mathbb{N} : r \mid P\}$. At resolution r, the model receives block-averaged history. Its forecast $\hat{\mathbf{y}}_t^{(r)}$ targets the P/r future block averages $A_r \mathbf{y}_t^{\text{fut}}$, where A_r is the future block-average operator (Section 4.3). Under a budget of at most K model calls per channel and origin, a procedure selects $\mathcal{S}_t \subseteq \mathcal{A}_P$ with $1 \in \mathcal{S}_t$ and $|\mathcal{S}_t| \leq K$. A combination rule $\mathcal{C}$ produces one fine-resolution forecast:

$$\tilde{\mathbf{y}}_t = \mathcal{C}\big(\{\hat{\mathbf{y}}_t^{(r)}\}_{r \in \mathcal{S}_t}, \{A_r\}_{r \in \mathcal{S}_t}, \mathcal{H}_t\big). \tag{3}$$

The goal is to minimize expected forecasting loss under these information and query constraints. Our experiments use squared error and the same evaluation origins and channels as the single-resolution baseline.

AMRI instantiates this setting with a fixed resolution set, $\mathcal{S}_t = \mathcal{R}$, and reconciliation weights estimated from pre-test calibration errors (CAL). The weights remain fixed throughout the test segment (TEST). The general setting permits varying resolution sets and feedback from completed TEST forecasts. Section 5 evaluates alternative selection and combination rules, including online baselines.

4 Method

4.1 Overview

AMRI queries the same frozen TSFM at multiple temporal resolutions and reconciles the outputs into a forecast at the original resolution. Each query uses a block-averaged history and predicts future block averages. Weighted least squares combines these forecasts into one prediction at the original sampling frequency, using weights estimated from historical forecast errors. Figure 2 shows calibration and inference. For multivariate data, AMRI calibrates and reconciles forecasts separately for each target channel.

4.2 Temporal Aggregation

At temporal resolution r, we average non-overlapping blocks of r consecutive observations. Let $D_{r,L}$ denote this transformation, retaining up to L block averages with the newest block ending at the forecast origin. In the default *expand* setting, L input points therefore represent Lr original observations. For example, $r = 4$ and $L = 512$ turn the most recent 2048 observations into 512 block averages. At $r = 1$, the model receives the original observations. We write D_r for $D_{r,L}$ when L is clear.

We query a fixed set of resolutions,

$$\mathcal{R} = \{1, 2, 4, 8\}, \tag{4}$$

with forecast horizons P divisible by 8, as in all main experiments. Each query predicts P/r block averages covering the same P future time steps. Appendix D compares alternative resolution sets and query budgets.

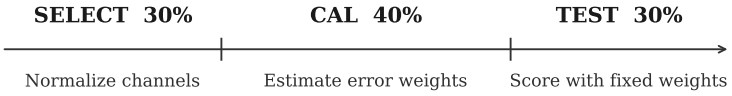

Figure 3: Chronological split: SELECT normalizes channels, CAL estimates weights, and TEST scores fixed weights.

The *matched* control averages the same last L observations at every resolution (Section 5). Optional covariates are transformed by C_r using the same block boundaries. Appendix B.2 specifies truncation, short-history handling, and covariate aggregation.

4.3 Forecasts of Future Block Averages

For each $r \in \mathcal{R}$, we query the same frozen forecasting function on the transformed history,

$$\hat{\mathbf{y}}_t^{(r)} = f_{\mathrm{FM}}(D_{r,L}(\mathbf{y}_{1:t}), C_r(\mathbf{x}_{1:t}); P/r) \in \mathbb{R}^{P/r}, \tag{5}$$

with the covariate argument omitted when covariates are not used.

Outputs at different resolutions predict different block averages, so they cannot be averaged directly.

Define the future block-average operator $A_r \in \mathbb{R}^{P/r \times P}$ by

$$\left(A_r \mathbf{y}_t^{\mathrm{fut}}\right)_j = \frac{1}{r} \sum_{i=(j-1)r+1}^{jr} y_{t+i}, \qquad j = 1, \ldots, P/r. \tag{6}$$

We call $\hat{\mathbf{y}}_t^{(r)}$ a native forecast because it predicts at the resolution of its input. Its target is $A_r \mathbf{y}_t^{\mathrm{fut}}$. For example, with $P = 8$ and $r = 2$, the four outputs predict the means of future steps 1–2, 3–4, 5–6, and 7–8. At the original resolution, $A_1 = I_P$, where $I_P \in \mathbb{R}^{P \times P}$ is the identity matrix. Thus D_r averages the observed input, while A_r defines the future block averages to be predicted.

Write $\mathcal{R} = \{r_1 < \cdots < r_{|\mathcal{R}|}\}$ for the queried set, ordered increasingly. Stacking the selected operators and native forecasts gives

$$A = \begin{bmatrix} A_{r_1} \\ \vdots \\ A_{r_{|\mathcal{R}|}} \end{bmatrix} \in \mathbb{R}^{n_{\mathcal{R}} \times P}, \qquad \hat{\mathbf{y}}_{\mathcal{R},t} = \begin{bmatrix} \hat{\mathbf{y}}_t^{(r_1)} \\ \vdots \\ \hat{\mathbf{y}}_t^{(r_{|\mathcal{R}|})} \end{bmatrix} \in \mathbb{R}^{n_{\mathcal{R}}}, \qquad n_{\mathcal{R}} = \sum_{r \in \mathcal{R}} \frac{P}{r}. \tag{7}$$

The rows of A define a temporal hierarchy: block averages of the same future at different resolutions (Athanasopoulos et al., 2017). Since $r_1 = 1$, the stack contains the identity block $A_{r_1} = I_P$, and hence A has full column rank.

4.4 Historical Reliability Calibration

We estimate the reliability of each forecast entry from its errors in CAL. Reliability is represented by statistical precision, the inverse forecast-error variance. Figure 3 summarizes the chronological split.

We estimate errors separately for each channel by replaying the multi-resolution queries at CAL origins with non-overlapping P-step forecast windows wholly inside CAL. The stacked error at origin t is

$$\mathbf{e}_t = \hat{\mathbf{y}}_{\mathcal{R},t} - A\mathbf{y}_t^{\mathrm{fut}} \in \mathbb{R}^{n_{\mathcal{R}}}. \tag{8}$$

Let $\mathbf{s}^2 \in \mathbb{R}^{n_{\mathcal{R}}}$ collect the entrywise empirical variances of $\mathbf{e}_t$ across CAL forecast origins (using the number of origins as the divisor), and let $\bar{s}^2$ denote the mean of the entries of $\mathbf{s}^2$. The default calibration precision matrix is

$$\widehat{\Lambda} = \left[\mathrm{diag}(\mathbf{s}^2) + \eta \bar{s}^2 I_{n_{\mathcal{R}}}\right]^{-1} \in \mathbb{R}^{n_{\mathcal{R}} \times n_{\mathcal{R}}}, \qquad \eta > 0, \tag{9}$$

where $I_{n_{\mathcal{R}}}$ is the identity matrix and $\eta > 0$ is a fixed dimensionless ridge constant. Each diagonal weight corresponds to one resolution and forecast position. The diagonal estimate avoids fitting

cross-entry covariances. The ridge limits large weights caused by small variances. If $\bar{s}^2 = 0$, we use $\widehat{\Lambda} = I_{n_{\mathcal{R}}}$. Centered variances measure error dispersion, so a constant bias does not reduce the weight. Table 3 evaluates a second-moment alternative that also accounts for bias.

4.5 Forecast Reconciliation

To extend the $P = 8$, $r = 2$ example in Section 4.3, suppose the original-resolution forecast averages to 10 over the first two future steps, while the $r = 2$ forecast predicts 12 for that same block. Reconciliation adjusts the fine-resolution values according to the relative reliability of these predictions.

We choose a forecast $\tilde{\mathbf{y}}_t \in \mathbb{R}^P$ whose block averages match the model outputs as closely as possible, weighted by their calibrated precision:

$$\tilde{\mathbf{y}}_t = \arg\min_{\mathbf{z}\in\mathbb{R}^P} (\hat{\mathbf{y}}_{\mathcal{R},t} - A\mathbf{z})^\top \widehat{\Lambda} (\hat{\mathbf{y}}_{\mathcal{R},t} - A\mathbf{z}).$$

Because A has full column rank and $\widehat{\Lambda} \succ 0$, the solution is unique and has the closed form

$$\tilde{\mathbf{y}}_t = \left(A^\top \widehat{\Lambda} A\right)^{-1} A^\top \widehat{\Lambda} \hat{\mathbf{y}}_{\mathcal{R},t}. \tag{10}$$

With diagonal $\widehat{\Lambda}$, this is weighted least squares (WLS) with inverse-variance weights (Hyndman et al., 2011). Using the full inverse error covariance gives generalized least squares (GLS), corresponding to minimum-trace (MinT) reconciliation (Wickramasuriya et al., 2019). For simplicity, TEST reuses the precomputed CAL matrix. Appendix J studies online weight updates.

4.6 Theoretical Analysis

Effect of a coarse forecast. For two resolutions, $\mathcal{R} = \{1, r_2\}$, write $\widehat{\Lambda} = \mathrm{blkdiag}(\widehat{\Lambda}^{(1)}, \widehat{\Lambda}^{(r_2)})$, where blkdiag forms a block-diagonal matrix and $\widehat{\Lambda}^{(r)} \in \mathbb{R}^{P/r\times P/r}$ is the precision block for resolution r. The reconciled forecast is

$$\tilde{\mathbf{y}}_t = \hat{\mathbf{y}}_t^{(1)} + W\left(\hat{\mathbf{y}}_t^{(r_2)} - A_{r_2}\hat{\mathbf{y}}_t^{(1)}\right), \qquad W = \left(\widehat{\Lambda}^{(1)} + A_{r_2}^\top \widehat{\Lambda}^{(r_2)} A_{r_2}\right)^{-1} A_{r_2}^\top \widehat{\Lambda}^{(r_2)} \in \mathbb{R}^{P\times P/r_2}. \tag{11}$$

The difference in parentheses is the coarse forecast minus block averages of the original-resolution forecast. The matrix W distributes this difference across the original time steps according to their precisions. Equal precisions within a block give a uniform correction.

If forecasting an aggregated history gives the same result as aggregating the original-resolution forecast (aggregation equivariance),

$$\hat{\mathbf{y}}_t^{(r)} = A_r \hat{\mathbf{y}}_t^{(1)} \qquad \text{for every } r \in \mathcal{R}, \tag{12}$$

then $\hat{\mathbf{y}}_{\mathcal{R},t} = A\hat{\mathbf{y}}_t^{(1)}$, and reconciliation returns $\tilde{\mathbf{y}}_t = \hat{\mathbf{y}}_t^{(1)}$ for any positive-definite $\widehat{\Lambda}$. Disagreement is therefore necessary to change the forecast.

Risk under exact error covariance. For the fixed resolution set $\mathcal{R}$, suppose

$$\hat{\mathbf{y}}_{\mathcal{R},t} = A\mathbf{y}_t^{\mathrm{fut}} + \boldsymbol{\varepsilon}_t, \qquad \mathbb{E}[\boldsymbol{\varepsilon}_t] = \mathbf{0}, \qquad \mathrm{Cov}(\boldsymbol{\varepsilon}_t) = \Sigma \succ 0,$$

where $\boldsymbol{\varepsilon}_t \in \mathbb{R}^{n_{\mathcal{R}}}$ is the stacked forecast error and $\Sigma \in \mathbb{R}^{n_{\mathcal{R}}\times n_{\mathcal{R}}}$ is its population covariance, assumed constant across origins.

Proposition 1 (Risk dominance under exact reliability)**.** *Under the measurement model above, if Equation* (10) *uses the exact population precision* Σ^{-1}*, then*

$$\begin{aligned} \mathrm{Cov}\left(\tilde{\mathbf{y}}_t - \mathbf{y}_t^{\mathrm{fut}}\right) &\preceq \mathrm{Cov}\left(\hat{\mathbf{y}}_t^{(1)} - \mathbf{y}_t^{\mathrm{fut}}\right), \\ \mathbb{E}\left\|\tilde{\mathbf{y}}_t - \mathbf{y}_t^{\mathrm{fut}}\right\|_2^2 &\le \mathbb{E}\left\|\hat{\mathbf{y}}_t^{(1)} - \mathbf{y}_t^{\mathrm{fut}}\right\|_2^2. \end{aligned} \tag{13}$$

This follows from the Gauss–Markov theorem: the original-resolution forecast is a linear unbiased estimator because A contains the identity block (Rao, 1973). The guarantee applies to the exact population precision Σ^{-1}, rather than the estimated diagonal $\widehat{\Lambda}$ used in AMRI. Appendix A gives the proof and the extensions to selected forecast positions and biased errors.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Models and data. We evaluate seven frozen TSFMs, Chronos-2 (Ansari et al., 2025), Chronos-Bolt (Amazon Web Services, 2024), TimesFM 2.5 (Google Research, 2025), TimesFM 3.0 (Das et al., 2024; Google Research, 2026), Moirai-2 (Liu et al., 2025a), Moirai-MoE (Liu et al., 2025b), and Sundial (Liu et al., 2025c), on eight benchmarks: ETTh1, ETTh2, ETTm1, ETTm2, exchange_rate, weather, electricity, and traffic. The horizons are $P \in \{24, 48, 96, 192\}$, giving 224 backbone–dataset–horizon combinations, which we call cells. To fit the compute budget, we use pre-specified channel subsets for three datasets: the target channel OT for weather, 20 load columns plus OT for electricity, and an evenly spaced 100 of 862 channels for traffic.

Protocol and metric. Each series is split chronologically into SELECT, CAL, and TEST in a $30/40/30$ ratio (Figure 3). Channels are standardized using their SELECT mean and standard deviation. Channels with zero SELECT standard deviation are left unscaled. CAL estimates reliability, which remains fixed during TEST. Calibration provides 15 to 1161 origins per channel, from exchange_rate at $P = 192$ to ETTm1 at $P = 24$. Forecast windows in CAL and TEST are non-overlapping P-step blocks. Non-overlap does not imply independent errors. We report $\mathrm{MSE}_{\mathrm{norm}}$, the mean squared error in these standardized units. Within each cell, errors are averaged over forecast steps and valid (origin, channel) pairs. Pooled results weight cells equally, and gain is $100(1 - \mathrm{MSE}_{\mathrm{method}}/\mathrm{MSE}_{\mathrm{base}})$, computed from the pooled errors.

Queries and comparisons. AMRI uses $\mathcal{R} = \{1, 2, 4, 8\}$ and $\eta = 10^{-2}$. Under the default *expand* setting, each query receives up to $L = 512$ aggregated points, so resolution r reads up to $512r$ original observations, capped by the available history. The single-resolution baseline reads the last 512 observations. A *matched* control averages the same last 512 observations at every resolution. Table 3 compares forecast-combination variants and adaptation methods. Appendices E and F specify them. We also compare DLinear (Zeng et al., 2023), PatchTST (Nie et al., 2023), and iTransformer (Liu et al., 2024), trained with three seeds each. These baselines use multivariate inputs and are scored on the same origins and channels as the frozen models. We consider training on SELECT+CAL and training on SELECT with CAL for early stopping. Appendix I gives the training protocol and dataset-level results.

5.2 FORECASTING ACCURACY

AMRI reduces pooled normalized MSE by 9.9% relative to the single-resolution baseline (Table 1), with positive average gains for every backbone (Table 3(b)). It improves on 185 of 224 cells. The one-sided Wilcoxon test over dataset-level aggregates gives $p = 0.02$. Figure 4 shows that improvements occur in most cells at every horizon, although their magnitude varies across models and datasets.

Table 1 shows the largest pooled dataset gain on ETTm1. Losses are concentrated on electricity and traffic, where both calibrated and identity-weighted reconciliation increase error. This pattern

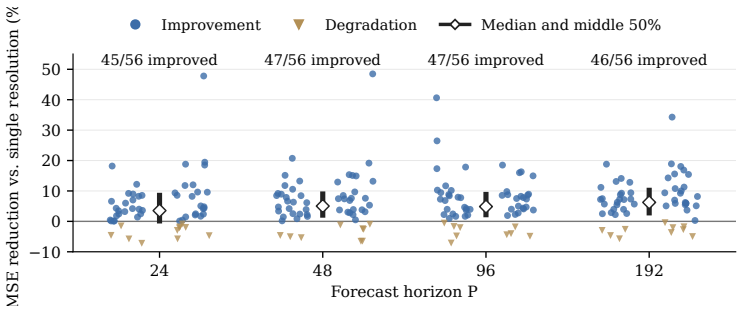

Figure 4: Per-cell gain with expanded history on 224 cells versus the 512-point baseline. Diamonds mark medians, bars interquartile ranges, and counts improved cells.

Table 1: Main forecasting results: normalized MSE averaged over seven backbones and four horizons per dataset (224 cells in total). Base uses one 512-point query. AMRI uses four queries with matched history (the same 512 observations) or expanded history (up to $512r$ observations at resolution r). Gains are relative to Base. Pooled errors weight all cells equally.

Dataset	Base	Matched	Expanded	Gain (M.)	Gain (E.)
ETTh1	0.9619	0.9367	0.9059	+2.6%	+5.8%
ETTh2	0.6497	0.6214	0.6081	+4.4%	+6.4%
ETTm1	0.9132	0.8059	0.7255	+11.8%	+20.6%
ETTm2	0.4406	0.4139	0.3961	+6.0%	+10.1%
exchange_rate	0.2251	0.2196	0.2135	+2.4%	+5.2%
weather	0.9197	0.8806	0.7909	+4.3%	+14.0%
electricity	0.2056	0.2134	0.2080	−3.8%	−1.2%
traffic	0.3223	0.3412	0.3296	−5.9%	−2.3%
Pooled	**0.5798**	**0.5541**	**0.5222**	**+4.4%**	**+9.9%**

Table 2: History length and calibration on the same 224 cells. Entries are pooled normalized MSE (percentage reduction from the single-resolution baseline, 0.5798). Both weighting rules reconcile queries at $r \in \{1, 2, 4, 8\}$. Matched history reads 512 observations at every resolution, while expanded history reads $512r$. CAL weights are regularized inverse error variances.

History	Equal weights	CAL weights (AMRI)
Matched	0.5567 (+4.0%)	0.5541 (+4.4%)
Expanded	0.5275 (+9.0%)	0.5222 (+9.9%)

suggests that changing calibration weights alone may be insufficient because averaging may remove useful short-term variation. Table 1 reports the dataset-level errors.

5.3 Contributions of History Length and Calibration

Table 2 crosses two history settings with two weighting rules. With matched history, equal-weight and CAL-weighted reconciliation improve over the baseline by 4.0% and 4.4%, respectively. With expanded history, the gains are 9.0% and 9.9%. Thus multiple temporal views help even when every query draws on the same observations. Calibration adds a modest improvement in both history settings, while expanding history yields a larger gain under both weighting rules. These contrasts describe empirical contributions without identifying a model-internal mechanism.

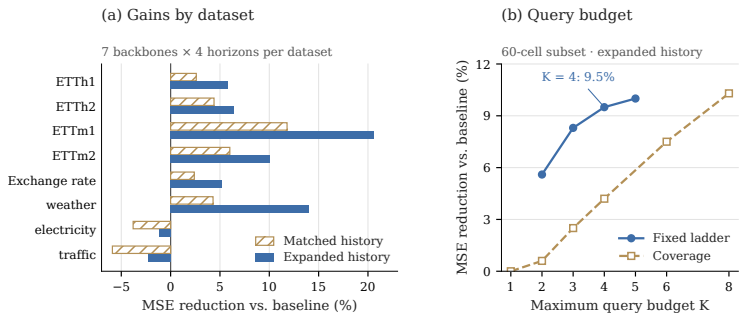

Figure 5: Normalized-MSE reduction versus the 512-point base. (a) Dataset gains across seven backbones and four horizons. Matched history reads 512 observations at each resolution, expanded history $512r$ (Table 1). (b) Gains on 60 cells versus maximum query budget K. The fixed ladder uses powers of two, and coverage uses variance retained after block averaging in SELECT (Table 9).

Table 3: Adaptation methods and backbone results on 224 cells. MSE is pooled normalized error. Gains are relative to the same-cell $r = 1$ base. The gradient head (*) covers 64 cells, and the affine oracle uses TEST labels. Appendix F specifies the methods and adaptation data.

(a) Methods and weighting variants

Method	Adapt. data	MSE	Gain	Cells
Base (r=1)	none	0.5798	0.0%	224
Equal weights, $\widehat{\Lambda}=I$	none	0.5275	9.0%	224
Heuristic shrinkage	CAL	0.5655	2.5%	224
Affine calibration	CAL	0.5692	1.8%	224
CtxRetrieval	SELECT+CAL	0.5712	1.5%	224
DriftDebias	CAL	0.5985	−3.2%	224
SMR^2 (Gao et al., 2026b)	CAL+TEST	0.5251	9.4%	224
ELF (Lee et al., 2025)	History+TEST	0.5322	8.2%	224
RGMR-inspired	CAL	0.5495	5.2%	224
ORCA (Dai et al., 2026)	CAL+TEST	0.5380	7.2%	224
DriftDebias (online)	CAL+TEST	0.6613	−14.1%	224
Gradient head	CAL+TEST	0.5851	−3.2%	64*
Affine oracle	TEST labels	0.4895	15.6%	224
AMRI, second moment	CAL	0.5209	10.2%	224
AMRI (ours)	CAL	0.5222	9.9%	224

(b) Across frozen backbones

Backbone	Base	AMRI	Gain
Chronos-2	0.5523	0.5018	+9.1%
Chronos-Bolt	0.5390	0.5166	+4.2%
TimesFM 2.5	0.5348	0.4922	+8.0%
TimesFM 3.0	0.5215	0.4807	+7.8%
Moirai-2	0.5812	0.5219	+10.2%
Moirai-MoE	0.8019	0.6433	+19.8%
Sundial	0.5275	0.4990	+5.4%

5.4 Comparison with Other Methods

With CAL weights fixed before TEST, AMRI achieves slightly lower pooled error than SMR^2 and ELF, which use feedback from earlier TEST forecasts (Table 3(a) and Appendix F). AMRI also has lower pooled error than reconciliation using the heuristic-shrinkage covariance estimate. Accounting for bias through second moments reduces error slightly further. Unconstrained ridge fitting on CAL increases pooled error even with its TEST-best regularization strength (Appendix E). Input transformations and step-specific affine corrections are examined in Appendices H and G.

Under SELECT-only training with CAL for early stopping, AMRI has lower pooled error than DLinear, PatchTST, and iTransformer. With SELECT+CAL training, PatchTST has slightly lower pooled error than AMRI (Appendix I, Table 13).

5.5 Query Budget and Long-Context Tradeoffs

On 60 cells, increasing the fixed ladder's query budget from four to five raises its gain from 9.5% to 10.0% (Figure 5(b) and Table 9). Channel-wise CAL selection reaches 12.4% with four TEST queries, or 12.1% with disjoint CAL blocks for selection and weighting (Appendix D, Table 8). On this grid, choosing resolutions improves accuracy more than extending the fixed ladder. This selection requires additional forecasts for candidate resolutions during calibration.

On the separate 128-cell subset covering four backbones, a single long-context query has lower pooled error than AMRI, whose error is 1.7% higher (Appendix D.1). Matched ETTh1 timings show that AMRI is slower than the 512-point base. Speed versus a long-context query depends on model and batch size (Appendix D.2). For Moirai-2, AMRI is faster at batch size eight but slower at batch size one. Query count alone is therefore insufficient to compare inference cost.

6 Conclusion

We presented AMRI, a training-free approach to improving frozen time-series foundation models through multi-resolution inference. AMRI queries the same model at different temporal resolutions and reconciles its forecasts using known aggregation relationships and reliability estimates calibrated before testing. Our analysis explains how cross-resolution disagreement adjusts the original-resolution forecast, while experiments across seven backbones and eight datasets demonstrate the benefits of forecast combination and broader history coverage. Together, these findings establish multi-resolution inference as a practical approach to using frozen TSFMs under query and input-length constraints.

Reproducibility Statement

Section 5 specifies the data splits, normalization, query settings, and evaluation metric. Appendix A gives the complete proof. Appendices C, E, F, and I report detailed results and baseline specifications. Appendix D.2 gives the timing protocol.

AI Use Statement

In this work, we used generative AI tools (Anthropic's Claude Code and OpenAI's Codex and image-generation tools) as an assistant in the research process, including helping implement and refactor experimental code, running and debugging experiments, providing feedback on experimental design and result interpretation, drafting and editing prose, refining figures, summarizing literature, and verifying references. All mathematical claims, derivations, and proofs were developed and verified by the authors, and all experimental results were produced by code reviewed and executed under the authors' control. We have reviewed all AI-assisted work. We take responsibility for the final content of this work.

References

Amazon Web Services. Chronos-Bolt: Fast and accurate zero-shot forecasting. Checkpoint `amazon/chronos-bolt-small` on Hugging Face, 2024. URL `https://huggingface.co/amazon/chronos-bolt-small`.

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Türkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025. URL `https://arxiv.org/abs/2510.15821`.

George Athanasopoulos, Rob J. Hyndman, Nikolaos Kourentzes, and Fotios Petropoulos. Forecasting with temporal hierarchies. *European Journal of Operational Research*, 262(1):60–74, 2017. doi: 10.1016/j.ejor.2017.02.046.

Peter Bloomfield. *Fourier Analysis of Time Series: An Introduction*. Wiley, New York, 2 edition, 2000. doi: 10.1002/0471722235.

Harold Felipe Calderón González and Alberto Gudiño-Ochoa. Temporal hierarchical forecast reconciliation of photovoltaic power generation from heterogeneous base models. *Measurement: Energy*, 10:100094, 2026. doi: 10.1016/j.meaene.2026.100094.

Defu Cao, Furong Jia, Sercan O. Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. TEMPO: Prompt-based generative pre-trained transformer for time series forecasting. In *International Conference on Learning Representations*, 2024.

Cristian Challu, Kin G. Olivares, Boris N. Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. NHITS: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023. doi: 10.1609/aaai.v37i6.25854.

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 *International Conference on Learning Representations*, 2024.

Xilin Dai, Yiding Liu, Hongjie Xia, Yifan Hu, Zewei Dong, Jiang-Ming Yang, and Qiang Xu. Learning the context of errors: Black-box online adaptation of time series foundation models. *arXiv preprint arXiv:2606.14222*, 2026. URL `https://arxiv.org/abs/2606.14222`.

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. PMLR, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

Tommaso Di Fonzo and Daniele Girolimetto. Cross-temporal forecast reconciliation: Optimal combination method and heuristic alternatives. *International Journal of Forecasting*, 39(1):39–57, 2023. doi: 10.1016/j.ijforecast.2021.08.004.

Vijay Ekambaram, Arindam Jati, Pankaj Dayama, Sumanta Mukherjee, Nam H. Nguyen, Wesley M. Gifford, Chandra Reddy, and Jayant Kalagnanam. Tiny time mixers (TTMs): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2359.

Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. UniTS: A unified multi-task time series model. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-4463.

Wentao Gao, Jiuyong Li, Lin Liu, Thuc Duy Le, Jixue Liu, Yanchang Zhao, and Yun Chen. Residual-guided multi-resolution refinement of foundation models: A case study in drought forecasting. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=ERCbhCrHo3.

Wentao Gao, Jiuyong Li, Lin Liu, Thuc Duy Le, Jixue Liu, Yanchang Zhao, and Yun Chen. Lightweight wrappers for adapting time series foundation models to regional drought forecasting. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '26, pp. 10957–10968, New York, NY, USA, 2026b. Association for Computing Machinery. ISBN 9798400722592. doi: 10.1145/3770855.3818835. URL https://doi.org/10.1145/3770855.3818835.

Google Research. TimesFM 2.5. Checkpoint google/timesfm-2.5-200m-pytorch on Hugging Face, 2025. URL https://huggingface.co/google/timesfm-2.5-200m-pytorch.

Google Research. TimesFM 3.0 (PyTorch). Checkpoint google/timesfm-3.0-pytorch on Hugging Face, 2026. URL https://huggingface.co/google/timesfm-3.0-pytorch.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 16115–16152. PMLR, 2024. URL https://proceedings.mlr.press/v235/goswami24a.html.

Ruijin Hua, Zichuan Liu, Kun Zhang, and Yiyuan Yang. Diversified scaling inference in time series foundation models. *arXiv preprint arXiv:2601.17376*, 2026. URL https://arxiv.org/abs/2601.17376.

Rob J. Hyndman, Roman A. Ahmed, George Athanasopoulos, and Han Lin Shang. Optimal combination forecasts for hierarchical time series. *Computational Statistics & Data Analysis*, 55(9): 2579–2589, 2011. doi: 10.1016/j.csda.2011.03.006.

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.

HyunGi Kim, Siwon Kim, Jisoo Mok, and Sungroh Yoon. Battling the non-stationarity in time series forecasting via test-time adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17868–17876, 2025. doi: 10.1609/aaai.v39i17.33965.

Nikolaos Kourentzes and George Athanasopoulos. Cross-temporal coherent forecasts for australian tourism. *Annals of Tourism Research*, 75:393–409, 2019. doi: 10.1016/j.annals.2019.02.001.

Thomas L. Lee, William Toner, Rajkarn Singh, Artjom Joosen, and Martin Asenov. Lightweight online adaption for time series foundation model forecasts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 33736–33764. PMLR, 2025. URL https://proceedings.mlr.press/v267/lee25ag.html.

Chenghao Liu, Taha Aksu, Juncheng Liu, Xu Liu, Hanshu Yan, Quang Pham, Silvio Savarese, Doyen Sahoo, Caiming Xiong, and Junnan Li. Moirai 2.0: When less is more for time series forecasting. *arXiv preprint arXiv:2511.11698*, 2025a. URL https://arxiv.org/abs/2511.11698.

Xu Liu, Juncheng Liu, Gerald Woo, Taha Aksu, Yuxuan Liang, Roger Zimmermann, Chenghao Liu, Junnan Li, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. Moirai-MoE: Empowering time series foundation models with sparse mixture of experts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 38940–38962. PMLR, 2025b. URL https://proceedings.mlr.press/v267/liu25an.html.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024.

Yong Liu, Guo Qin, Zhiyuan Shi, Zhi Chen, Caiyin Yang, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Sundial: A family of highly capable time series foundation models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 39295–39317. PMLR, 2025c. URL https://proceedings.mlr.press/v267/liu25be.html.

Heitor R. Medeiros, Hossein Sharifi-Noghabi, Gabriel L. Oliveira, and Saghar Irandoust. Accurate parameter-efficient test-time adaptation for time series forecasting. In *2nd Workshop on Test-Time Adaptation: Putting Updates to the Test! at ICML 2025*, 2025. URL https://openreview.net/pdf?id=uFj4EL4GTB.

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

Syama Sundar Rangapuram, Lucien D. Werner, Konstantinos Benidis, Pedro Mercado, Jan Gasthaus, and Tim Januschowski. End-to-end learning of coherent probabilistic forecasts for hierarchical time series. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8832–8843. PMLR, 2021. URL https://proceedings.mlr.press/v139/rangapuram21a.html.

C. Radhakrishna Rao. *Linear Statistical Inference and its Applications*. Wiley, New York, 2 edition, 1973. doi: 10.1002/9780470316436.

Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Hena Ghonia, Rishika Bhagwatkar, Arian Khorasani, Mohammad Javad Darvishi Bayazi, George Adamopoulos, Roland Riachi, Nadhir Hassen, et al. Lag-llama: Towards foundation models for probabilistic time series forecasting. *arXiv preprint arXiv:2310.08278*, 2024. URL https://arxiv.org/abs/2310.08278.

Mohammad Amin Shabani, Amir H. Abdi, Lili Meng, and Tristan Sylvain. Scaleformer: Iterative multi-scale refining transformers for time series forecasting. In *International Conference on Learning Representations*, 2023.

Evangelos Spiliotis, Mahdi Abolghasemi, Rob J. Hyndman, Fotios Petropoulos, and Vassilios Assimakopoulos. Hierarchical forecast reconciliation with machine learning. *Applied Soft Computing*, 112:107756, 2021. doi: 10.1016/j.asoc.2021.107756.

Asterios Tsiourvas, Wei Sun, Georgia Perakis, Pin-Yu Chen, and Yada Zhu. Learning optimal projection for forecast reconciliation of hierarchical time series. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 48713–48727. PMLR, 2024. URL https://proceedings.mlr.press/v235/tsiourvas24b.html.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024.

Shiyu Wang, Wei Lu, Jiawei Li, Xiaoming Shi, Xinyue Zhong, Zhou Ye, Ming Jin, and Qingsong Wen. FRT: Flow-based reconcile transformer for hierarchical time series. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 4995–5006, 2025. doi: 10.1145/3711896.3737224.

Shanika L. Wickramasuriya, George Athanasopoulos, and Rob J. Hyndman. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association*, 114(526):804–819, 2019. doi: 10.1080/01621459.2018.1448825.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53140–53164. PMLR, 2024. URL https://proceedings.mlr.press/v235/woo24a.html.

Congxi Xiao, Jingbo Zhou, Yixiong Xiao, Xinjiang Lu, Le Zhang, and Hui Xiong. TimeFound: A foundation model for time series forecasting. *arXiv preprint arXiv:2503.04118*, 2025. URL https://arxiv.org/abs/2503.04118.

Hao Xue and Flora D. Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6851–6864, 2024. doi: 10.1109/TKDE.2023.3342137.

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.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 2022. URL https://proceedings.mlr.press/v162/zhou22g.html.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained LM. In *Advances in Neural Information Processing Systems*, volume 36, pp. 43322–43355, 2023. doi: 10.52202/075280-1877.

A Proof of Proposition 1

The covariance assumption $\Sigma \succ 0$ excludes exact aggregation equivariance when multiple resolutions are stacked: in that case the errors lie in the P-dimensional column space of A, so their covariance is singular. The linear aggregation relations do not require the forecasting model itself to be linear.

With the exact population precision Σ^{-1}, the normal equations for the weighted least-squares objective in Section 4.5 are

$$A^\top \Sigma^{-1} A\, \tilde{\mathbf{y}}_t = A^\top \Sigma^{-1} \hat{\mathbf{y}}_{\mathcal{R},t}. \tag{14}$$

The identity block makes A full column rank, so $A^\top \Sigma^{-1} A$ is positive definite. Hence

$$\tilde{\mathbf{y}}_t = \Pi_{\mathrm{GLS}} \hat{\mathbf{y}}_{\mathcal{R},t}, \qquad \Pi_{\mathrm{GLS}} = (A^\top \Sigma^{-1} A)^{-1} A^\top \Sigma^{-1}. \tag{15}$$

Π_{GLS} is the linear reconciliation map.

Let Π_{base} denote the linear map that extracts the $r = 1$ block of the stacked vector, that is,

$$\Pi_{\mathrm{base}} = [I_P \quad 0]\,. \tag{16}$$

Under the stacking order of Equation (7), $\hat{\mathbf{y}}_t^{(1)} = \Pi_{\mathrm{base}} \hat{\mathbf{y}}_{\mathcal{R},t}$. Both $\Pi_{\mathrm{GLS}} A = I_P$ and $\Pi_{\mathrm{base}} A = I_P$, so both estimators are unbiased under the measurement model in Section 4.6. Define

$$\Delta = \Pi_{\mathrm{base}} - \Pi_{\mathrm{GLS}}. \tag{17}$$

Then $\Delta A = 0$. Moreover, $\Pi_{\mathrm{GLS}}\Sigma = (A^\top\Sigma^{-1}A)^{-1}A^\top$, which gives $\Pi_{\mathrm{GLS}}\Sigma\Delta^\top = 0$. Therefore,

$$\begin{aligned}&\mathrm{Cov}(\Pi_{\mathrm{base}}\hat{\mathbf{y}}_{\mathcal{R},t} - \mathbf{y}_t^{\mathrm{fut}}) - \mathrm{Cov}(\Pi_{\mathrm{GLS}}\hat{\mathbf{y}}_{\mathcal{R},t} - \mathbf{y}_t^{\mathrm{fut}}) \\ &\quad = \Delta\Sigma\Delta^\top \succeq 0.\end{aligned} \tag{18}$$

This proves the covariance inequality in Equation (13). Taking the trace gives the squared-error inequality. For the evaluated-subset extension, let $J \in \mathbb{R}^{m\times P}$ be a selection matrix for m evaluated positions. Premultiplying and postmultiplying Equation (18) by J and $J^\top$, then taking the trace, gives

$$\mathbb{E}\Big[\big\|J(\tilde{\mathbf{y}}_t - \mathbf{y}_t^{\mathrm{fut}})\big\|_2^2\Big] \le \mathbb{E}\left[\left\|J(\hat{\mathbf{y}}_t^{(1)} - \mathbf{y}_t^{\mathrm{fut}})\right\|_2^2\right]. \tag{19}$$

Strict improvement holds exactly when $\mathrm{tr}(J\Delta\Sigma\Delta^\top J^\top) > 0$.

The unbiasedness assumption is essential. If $\mathbb{E}[\boldsymbol{\varepsilon}_t] = \mathbf{b} \neq \mathbf{0}$ while $\mathrm{Cov}(\boldsymbol{\varepsilon}_t) = \Sigma$ remains unchanged, then for any linear reconciliation map Π satisfying $\Pi A = I_P$,

$$\mathbb{E}\Big[\big\|\Pi\hat{\mathbf{y}}_{\mathcal{R},t} - \mathbf{y}_t^{\mathrm{fut}}\big\|_2^2\Big] = \|\Pi\mathbf{b}\|_2^2 + \mathrm{tr}(\Pi\Sigma\Pi^\top). \tag{20}$$

GLS minimizes the covariance contribution among linear maps satisfying $\Pi A = I_P$, but does not generally minimize the additional squared-bias term. The population dominance guarantee therefore does not automatically extend to biased measurements.

B Variation-Coverage Resolution Selection

This appendix specifies the data-dependent alternative to the fixed ladder of Section 4.2, and the sweep that compares the two. Let $\mathbf{y}^{\mathrm{SELECT}}$ denote the longest right-aligned suffix of SELECT whose length N is divisible by P. For selection we aggregate this common suffix without the input-length budget of Section 4.2: D_r^{sel} averages the full $\mathbf{y}^{\mathrm{SELECT}}$ over non-overlapping blocks of size r, so every resolution consumes the same observations. For each admissible $r \mid P$, the variation-retention ratio is

$$q_r = \frac{\widehat{\mathrm{Var}}\big(D_r^{\mathrm{sel}}(\mathbf{y}^{\mathrm{SELECT}})\big)}{\widehat{\mathrm{Var}}(\mathbf{y}^{\mathrm{SELECT}})}, \qquad r \mid P, \tag{21}$$

with $\widehat{\mathrm{Var}}$ the empirical second central moment. It measures the fraction of empirical variation retained after resolution-r aggregation, which describes how much each block size smooths the series. It is computed from the target series in SELECT alone and depends on neither covariates nor forecasting performance. The subsections below give its finite-sample decomposition and spectral interpretation.

B.1 Deterministic Coverage Selection

The rule places K evenly spaced retention targets over the observed range of q_r and matches each to the nearest admissible resolution, selecting views with different amounts of retained variance. The common suffix $\mathbf{y}^{\mathrm{SELECT}}$ contains complete blocks for every $r \mid P$, which prevents candidate-specific truncation from changing the data used to compute q_r. We require $N \ge 4P$ and nonzero empirical variance. The first condition ensures at least four aggregated observations even for the coarsest admissible resolution $r = P$. If either condition fails, we fall back to $\mathcal{R} = \{1\}$, reducing the procedure to original-resolution zero-shot inference.

This selection variant uses a feasible K, so K does not exceed the number of admissible resolutions. If a larger value is supplied in implementation, all admissible resolutions are returned. When $K = 1$, we set $\mathcal{R} = \{1\}$. For a feasible requested budget $K \ge 2$, define

$$q_{\min} = \min_{r\mid P} q_r, \qquad \tau_k = 1 - \frac{k-1}{K-1}(1 - q_{\min}), \qquad k = 1, \dots, K. \tag{22}$$

Each target is matched to the admissible resolution with the nearest retention value,

$$\tilde{r}_k = \min \arg\min_{r\mid P} |q_r - \tau_k|, \tag{23}$$

where the outer minimum resolves ties in favor of the smaller resolution. Because $q_1 = 1$, this rule retains $r = 1$. Let $\widetilde{\mathcal{R}} = \{\tilde{r}_k\}_{k=1}^{K}$ after removing duplicates. If $|\widetilde{\mathcal{R}}| < K$, we repeatedly add

$$r^\star = \min \arg\max_{\substack{r|P \\ r \notin \widetilde{\mathcal{R}}}} \min_{s \in \widetilde{\mathcal{R}}} |q_r - q_s|, \tag{24}$$

again resolving ties toward the smaller resolution. This addition of the most distant retention value continues until $|\widetilde{\mathcal{R}}| = K$. The resulting set is then sorted by resolution and relabeled as $\mathcal{R} = \{1 = r_1 < \cdots < r_K\}$, taking the place of the fixed ladder of Equation (4) in this variant.

B.2 History and Covariate Aggregation

History windows. To compute $D_{r,L}$, we keep the last at most Lr observations, discard the oldest points needed to make the retained length divisible by r, and average the resulting non-overlapping blocks. This aligns the newest block with the forecast origin. If fewer than Lr observations are available, the model receives fewer than L block averages. For the matched control, $D_{r,\lfloor L/r \rfloor}$ retains the last $r\lfloor L/r \rfloor$ original observations. For the fixed ladder, every queried resolution divides $L = 512$, so the matched input contains L/r averages of exactly the same last L observations. Both settings reduce to the last-L window at $r = 1$.

Resolution set. The fixed set uses successive powers of two, following multi-resolution designs in temporal-aggregation forecasting (Athanasopoulos et al., 2017), multi-rate architectures (Shabani et al., 2023), and multi-resolution foundation models (Xiao et al., 2025). The default uses all four resolutions in Equation (4) with P divisible by 8, ensuring integer output lengths. The data-dependent alternative is defined in Appendix B.

Covariates. C_r uses the same right-aligned truncation and block boundaries as the target transformation D_r, ensuring temporal alignment between the target and covariate views. Continuous state or intensity variables, such as temperature, are aggregated by block averages. Flow variables representing quantities accumulated over time, such as precipitation, are aggregated by block sums to preserve their accumulated physical meaning over the coarser interval. Static variables are carried unchanged across resolutions. Other categorical or application-specific covariates use a deterministic aggregation rule appropriate to their semantics. Before transformation, the target and covariates are aligned on their shared timestamps and restricted to the same right-aligned suffix. The retention ratio in Equation (21) is computed from the target series alone. At $r = 1$, C_1 retains the same last L observations as $D_{1,L}$ without averaging.

B.3 Finite-Sample Interpretation

Throughout, $\widehat{\mathrm{Var}}$ is the empirical second central moment: for a vector $x = [x_1, \ldots, x_n]^\top$ with mean $\bar{x}$, $\widehat{\mathrm{Var}}(x) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$. We next give an exact finite-sample interpretation of Equation (21). Let the common SELECT suffix contain $N = mr$ observations, where $m = N/r$ is the number of resolution-r blocks. Write $y_{j,i}$ for observation i in block j, let $\bar{y}_j^{(r)} = r^{-1}\sum_{i=1}^{r} y_{j,i}$ be the corresponding block mean, and let $\bar{y}$ be the mean of all N observations. The total sum-of-squares decomposition gives

$$\begin{aligned}\widehat{\mathrm{Var}}(\mathbf{y}^{\mathrm{SELECT}}) = \frac{1}{N}\sum_{j=1}^{m}\sum_{i=1}^{r}\left(y_{j,i} - \bar{y}_j^{(r)}\right)^2 \\ + \frac{1}{m}\sum_{j=1}^{m}\left(\bar{y}_j^{(r)} - \bar{y}\right)^2.\end{aligned} \tag{25}$$

The second term is exactly $\widehat{\mathrm{Var}}(D_r^{\mathrm{sel}}(\mathbf{y}^{\mathrm{SELECT}}))$, while the first is the empirical variation within resolution-r blocks. Consequently, q_r is the fraction of total empirical variation remaining between blocks after aggregation. Both terms in Equation (25) are nonnegative, so $0 \le q_r \le 1$, and the singleton partition gives $q_1 = 1$. This identity applies to the observed SELECT suffix without any stationarity assumption.

B.4 Spectral Interpretation

For a weakly stationary process with a spectral density and positive variance, the population counterpart admits a frequency-domain interpretation (Bloomfield, 2000). Writing i for the imaginary unit and $\omega \in (-\pi, \pi]$ for frequency, the length-r block average is a linear filter with frequency response

$$H_r(\omega) = \frac{1}{r} \sum_{\ell=1}^{r} e^{\mathrm{i}\omega\ell} = \frac{\sin(\omega r/2)}{r \, \sin(\omega/2)} \, e^{\mathrm{i}\omega(r+1)/2}. \tag{26}$$

It satisfies $|H_r(0)| = 1$, and $H_r(\omega)$ vanishes at the nonzero frequencies congruent to $2\pi k/r$ modulo 2π, for $k = 1, \ldots, r-1$. Thus aggregation preserves the zero-frequency component while attenuating or eliminating higher-frequency variation.

Let $S(\omega)$ denote the spectral density and let $\gamma(h) = \mathrm{Cov}(y_t, y_{t+h})$ denote the autocovariance at lag h. Throughout this subsection we use the spectral-density normalization

$$\gamma(h) = \int_{-\pi}^{\pi} S(\omega) e^{\mathrm{i}\omega h} \, \mathrm{d}\omega, \tag{27}$$

so that $\gamma(0)$ equals the integral of S over $(-\pi, \pi]$. The population variance of the block-averaged process satisfies

$$\mathrm{Var}\left(\frac{1}{r} \sum_{i=1}^{r} y_{t+i} \right) = \frac{1}{r^2} \sum_{i=1}^{r} \sum_{j=1}^{r} \gamma(i-j) = \int_{-\pi}^{\pi} |H_r(\omega)|^2 S(\omega) \, \mathrm{d}\omega. \tag{28}$$

Consequently, the population counterpart of Equation (21) is

$$q_r^{\mathrm{pop}} = \frac{\int_{-\pi}^{\pi} |H_r(\omega)|^2 S(\omega) \, \mathrm{d}\omega}{\int_{-\pi}^{\pi} S(\omega) \, \mathrm{d}\omega}. \tag{29}$$

Variation coverage therefore has both an exact empirical decomposition and a spectral-filtering interpretation, although its operational definition and implementation require only block averaging and empirical variances.

C Full Test-Segment Results

Table 1 summarizes dataset-level errors. Tables 4 through 7 list the $\mathrm{MSE}_{\mathrm{norm}}$ behind the aggregates of Section 5, per backbone, dataset, and horizon, on the same 224 cells and the same origins. Each row carries the single-resolution base and the deployed reconciliation with its relative gain, and each table names the backbones it holds.

Table 4: Full test-segment results, $\mathrm{MSE}_{\mathrm{norm}}$ per dataset and horizon under the deployed configuration (Chronos-2, part 1 of 4).

Backbone	Dataset	P	Base	AMRI	Gain
Chronos-2	ETTh1	24	0.7257	0.6909	+4.8%
		48	0.8712	0.8050	+7.6%
		96	1.1249	1.0244	+8.9%
		192	1.4243	1.2926	+9.2%
	ETTh2	24	0.4058	0.3912	+3.6%
		48	0.5591	0.5208	+6.9%
		96	0.8144	0.7346	+9.8%
		192	0.9922	0.8376	+15.6%
	ETTm1	24	0.5003	0.4413	+11.8%
		48	0.6966	0.6047	+13.2%
		96	0.8292	0.7323	+11.7%
		192	0.9470	0.8448	+10.8%
	ETTm2	24	0.2827	0.2594	+8.2%
		48	0.3437	0.3145	+8.5%
		96	0.4514	0.4126	+8.6%
		192	0.6180	0.5600	+9.4%
	exchange_rate	24	0.0616	0.0615	+0.1%
		48	0.1207	0.1178	+2.4%
		96	0.1864	0.1834	+1.7%
		192	0.4652	0.4519	+2.9%
	weather	24	0.3930	0.3452	+12.2%
		48	0.6358	0.5536	+12.9%
		96	0.9332	0.7936	+15.0%
		192	1.2450	1.0693	+14.1%
	electricity	24	0.1615	0.1577	+2.4%
		48	0.1809	0.1754	+3.0%
		96	0.2133	0.2055	+3.7%
		192	0.2514	0.2421	+3.7%
	traffic	24	0.2763	0.2758	+0.2%
		48	0.2982	0.2967	+0.5%
		96	0.3216	0.3184	+1.0%
		192	0.3437	0.3428	+0.3%

C.1 Forecast Examples

Figure 6 shows two TEST windows for Moirai-2 at $P = 96$. On ETTm1, AMRI moves the forecast closer to the observations near the end of the horizon, reducing window MSE by 16.2%. On electricity, the two forecasts largely overlap, but AMRI increases window MSE by 2.0%.

D Marginal Returns of the Query Budget

The subset enumeration runs on a fixed subgrid of the main experiment rather than all 224 cells: three backbones (Chronos-2, Moirai-2, TimesFM 3.0), five datasets (ETTh1, ETTh2, exchange_rate, weather, electricity), and all four horizons, expand span, giving $3 \times 5 \times 4 = 60$ cells. The 60 cells retain the origins and evaluated channels of the main experiment.

Table 8 compares resolution-selection rules. The fixed ladder follows geometric multi-scale designs (Shabani et al., 2023; Xiao et al., 2025). Uniform index spacing selects evenly spaced entries from the sorted admissible resolutions. Coverage selection uses SELECT variance to choose resolutions, whereas learned multi-scale routers such as Pathformer (Chen et al., 2024) select patch sizes through training. Five random subsets provide a reference for variability across choices: pooled $\mathrm{MSE}_{\mathrm{norm}}$ ranges from 0.5220 to 0.5634, with a mean of 0.5407.

Table 5: Full test-segment results, continued (Chronos-Bolt and TimesFM 2.5, part 2 of 4).

Backbone	Dataset	P	Base	AMRI	Gain
Chronos-Bolt	ETTh1	24	0.7375	0.7344	+0.4%
		48	0.8529	0.8390	+1.6%
		96	1.0366	0.9980	+3.7%
		192	1.2878	1.2366	+4.0%
	ETTh2	24	0.4003	0.3919	+2.1%
		48	0.5313	0.5133	+3.4%
		96	0.7121	0.6985	+1.9%
		192	0.8056	0.7644	+5.1%
	ETTm1	24	0.5697	0.5012	+12.0%
		48	0.7798	0.6618	+15.1%
		96	0.8715	0.7828	+10.2%
		192	0.9454	0.8794	+7.0%
	ETTm2	24	0.2770	0.2576	+7.0%
		48	0.3389	0.3122	+7.9%
		96	0.4423	0.4051	+8.4%
		192	0.5957	0.5397	+9.4%
	exchange_rate	24	0.0636	0.0620	+2.5%
		48	0.1216	0.1189	+2.2%
		96	0.1761	0.1718	+2.4%
		192	0.3742	0.3843	-2.7%
	weather	24	0.4175	0.4250	-1.8%
		48	0.5733	0.5571	+2.8%
		96	0.9466	0.9079	+4.1%
		192	1.3002	1.2262	+5.7%
	electricity	24	0.1667	0.1703	-2.1%
		48	0.1858	0.1876	-1.0%
		96	0.2120	0.2156	-1.7%
		192	0.2408	0.2480	-3.0%
	traffic	24	0.2817	0.2979	-5.8%
		48	0.3039	0.3202	-5.4%
		96	0.3316	0.3450	-4.0%
		192	0.3667	0.3763	-2.6%
TimesFM 2.5	ETTh1	24	0.7636	0.7178	+6.0%
		48	0.8907	0.8243	+7.5%
		96	1.0311	0.9429	+8.5%
		192	1.1294	1.0628	+5.9%
	ETTh2	24	0.4121	0.3948	+4.2%
		48	0.5477	0.5186	+5.3%
		96	0.7347	0.7006	+4.6%
		192	0.8064	0.7404	+8.2%
	ETTm1	24	0.5102	0.4683	+8.2%
		48	0.6531	0.5976	+8.5%
		96	0.7823	0.7214	+7.8%
		192	0.8819	0.8298	+5.9%
	ETTm2	24	0.2869	0.2624	+8.5%
		48	0.3493	0.3180	+8.9%
		96	0.4374	0.4075	+6.8%
		192	0.5855	0.5429	+7.3%
	exchange_rate	24	0.0625	0.0622	+0.4%
		48	0.1223	0.1175	+3.9%
		96	0.2061	0.1958	+5.0%
		192	0.5035	0.4761	+5.4%
	weather	24	0.4609	0.3713	+19.5%
		48	0.6886	0.5828	+15.4%
		96	0.9463	0.7774	+17.9%
		192	1.2464	1.0213	+18.1%
	electricity	24	0.1685	0.1700	-0.8%
		48	0.1912	0.1895	+0.9%
		96	0.2228	0.2191	+1.6%
		192	0.2585	0.2527	+2.2%
	traffic	24	0.2811	0.2895	-3.0%
		48	0.3006	0.3085	-2.6%
		96	0.3196	0.3254	-1.8%
		192	0.3331	0.3400	-2.1%

Table 6: Full test-segment results, continued (TimesFM 3.0 and Moirai-2, part 3 of 4).

Backbone	Dataset	P	Base	AMRI	Gain
TimesFM 3.0	ETTh1	24	0.7140	0.6832	+4.3%
		48	0.8576	0.8006	+6.6%
		96	1.0071	0.9314	+7.5%
		192	1.1186	1.0574	+5.5%
	ETTh2	24	0.4014	0.3875	+3.5%
		48	0.5336	0.5083	+4.7%
		96	0.7176	0.6891	+4.0%
		192	0.8138	0.7384	+9.3%
	ETTm1	24	0.4521	0.4222	+6.6%
		48	0.5919	0.5670	+4.2%
		96	0.7190	0.6902	+4.0%
		192	0.8506	0.8023	+5.7%
	ETTm2	24	0.2808	0.2555	+9.0%
		48	0.3410	0.3097	+9.2%
		96	0.4389	0.4040	+7.9%
		192	0.5963	0.5517	+7.5%
	exchange_rate	24	0.0643	0.0628	+2.4%
		48	0.1333	0.1251	+6.1%
		96	0.2225	0.2042	+8.2%
		192	0.5734	0.5093	+11.2%
	weather	24	0.4174	0.3415	+18.2%
		48	0.6629	0.5628	+15.1%
		96	0.9207	0.7504	+18.5%
		192	1.2088	0.9803	+18.9%
	electricity	24	0.1660	0.1659	+0.1%
		48	0.1878	0.1850	+1.4%
		96	0.2216	0.2166	+2.3%
		192	0.2550	0.2486	+2.5%
	traffic	24	0.2740	0.2781	-1.5%
		48	0.2941	0.2976	-1.2%
		96	0.3168	0.3186	-0.6%
		192	0.3368	0.3382	-0.4%
Moirai-2	ETTh1	24	0.7115	0.6815	+4.2%
		48	0.8561	0.7927	+7.4%
		96	1.0533	0.9451	+10.3%
		192	1.2302	1.0915	+11.3%
	ETTh2	24	0.4151	0.3984	+4.0%
		48	0.5434	0.5223	+3.9%
		96	0.7550	0.7231	+4.2%
		192	0.8127	0.7544	+7.2%
	ETTm1	24	0.6048	0.4911	+18.8%
		48	0.8216	0.6512	+20.7%
		96	0.9226	0.7723	+16.3%
		192	1.0311	0.8837	+14.3%
	ETTm2	24	0.3017	0.2734	+9.4%
		48	0.3651	0.3298	+9.7%
		96	0.4647	0.4237	+8.8%
		192	0.6611	0.5762	+12.8%
	exchange_rate	24	0.0614	0.0603	+1.7%
		48	0.1245	0.1199	+3.8%
		96	0.2050	0.1953	+4.7%
		192	0.4713	0.4417	+6.3%
	weather	24	0.5156	0.4661	+9.6%
		48	0.7458	0.6345	+14.9%
		96	1.2004	0.9925	+17.3%
		192	1.6889	1.3713	+18.8%
	electricity	24	0.1674	0.1707	-2.0%
		48	0.1839	0.1884	-2.4%
		96	0.2127	0.2171	-2.1%
		192	0.2457	0.2501	-1.8%
	traffic	24	0.2752	0.2882	-4.7%
		48	0.2943	0.3092	-5.1%
		96	0.3180	0.3320	-4.4%
		192	0.3396	0.3521	-3.7%

Table 7: Full test-segment results, continued (Moirai-MoE and Sundial, part 4 of 4).

Backbone	Dataset	P	Base	AMRI	Gain
Moirai-MoE	ETTh1	24	0.8145	0.7883	+3.2%
		48	0.9284	0.9049	+2.5%
		96	1.1128	1.0645	+4.3%
		192	1.3346	1.2517	+6.2%
	ETTh2	24	0.4539	0.4308	+5.1%
		48	0.6185	0.5715	+7.6%
		96	0.8843	0.8007	+9.5%
		192	0.9982	0.8673	+13.1%
	ETTm1	24	1.6160	0.8436	+47.8%
		48	2.3681	1.2199	+48.5%
		96	1.9635	1.1658	+40.6%
		192	1.9102	1.2555	+34.3%
	ETTm2	24	0.3900	0.3178	+18.5%
		48	0.4713	0.3810	+19.1%
		96	0.5852	0.4916	+16.0%
		192	0.7760	0.6562	+15.4%
	exchange_rate	24	0.0617	0.0616	+0.1%
		48	0.1171	0.1169	+0.2%
		96	0.1856	0.1817	+2.1%
		192	0.5142	0.4690	+8.8%
	weather	24	0.6350	0.5766	+9.2%
		48	0.8481	0.7357	+13.3%
		96	1.7402	1.2796	+26.5%
		192	1.9459	1.6177	+16.9%
	electricity	24	0.2021	0.2138	-5.8%
		48	0.2254	0.2401	-6.6%
		96	0.2591	0.2718	-4.9%
		192	0.2937	0.3084	-5.0%
	traffic	24	0.3043	0.3262	-7.2%
		48	0.3374	0.3594	-6.5%
		96	0.3652	0.3913	-7.1%
		192	0.4005	0.4234	-5.7%
Sundial	ETTh1	24	0.6644	0.6547	+1.5%
		48	0.7750	0.7455	+3.8%
		96	0.8765	0.8419	+3.9%
		192	1.0015	0.9620	+3.9%
	ETTh2	24	0.4083	0.3982	+2.5%
		48	0.5414	0.5242	+3.2%
		96	0.7276	0.7113	+2.2%
		192	0.8457	0.7953	+6.0%
	ETTm1	24	0.4772	0.4313	+9.6%
		48	0.6411	0.5734	+10.6%
		96	0.7547	0.6885	+8.8%
		192	0.8792	0.7912	+10.0%
	ETTm2	24	0.2871	0.2626	+8.6%
		48	0.3510	0.3213	+8.4%
		96	0.4389	0.4067	+7.3%
		192	0.5779	0.5363	+7.2%
	exchange_rate	24	0.0688	0.0659	+4.3%
		48	0.1426	0.1336	+6.3%
		96	0.2680	0.2470	+7.8%
		192	0.6251	0.5793	+7.3%
	weather	24	0.4301	0.4223	+1.8%
		48	0.6949	0.6131	+11.8%
		96	1.0261	0.9514	+7.3%
		192	1.2845	1.2192	+5.1%
	electricity	24	0.1400	0.1464	-4.6%
		48	0.1557	0.1630	-4.6%
		96	0.1807	0.1894	-4.8%
		192	0.2059	0.2155	-4.7%
	traffic	24	0.3089	0.3046	+1.4%
		48	0.3401	0.3325	+2.2%
		96	0.3682	0.3581	+2.7%
		192	0.3931	0.3829	+2.6%

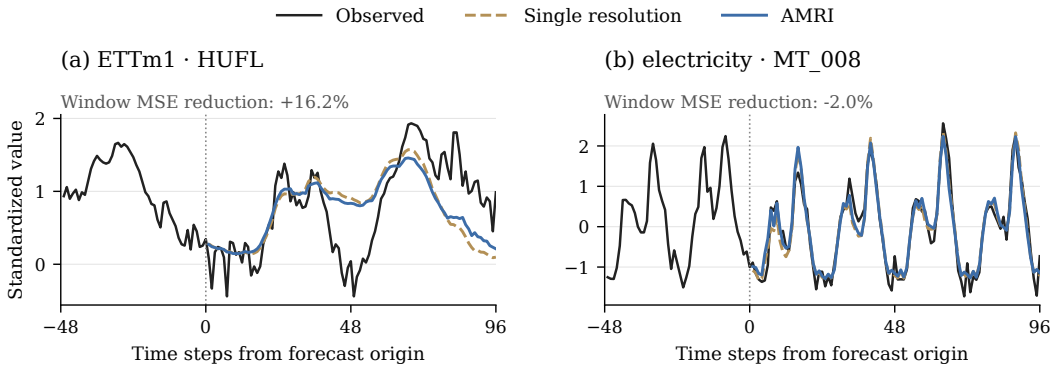

Figure 6: Forecast examples for Moirai-2 with expanded history and $P = 96$. The dotted line marks the forecast origin. Values use SELECT standardization. For each dataset, we select a TEST window and channel whose MSE reduction is closest to the corresponding cell's pooled reduction, among pairs with baseline MSE in the middle 50% and the same gain sign. Displayed reductions use all 96 future steps. The preceding 48 observations provide context.

Table 8: Resolution-selection rules on the 60 cells of the three-backbone, five-dataset grid. $\mathrm{MSE}_{\mathrm{norm}}$ is the mean of the 60 cell-level errors. Queries is the number of backbone calls one forecast origin costs (Appendix D.2). The base uses one query, all-admissible uses every candidate resolution, and all other rows use $K = 4$ queries. The admissible set $\mathcal{A}_P$ has 8, 10, 12, or 14 resolutions at $P = 24, 48, 96, 192$, so all-admissible costs 8 to 14 calls (mean 11). Percentile is computed per cell: the rule's cell-level $\mathrm{MSE}_{\mathrm{norm}}$ is ranked within that cell's own error distribution over all admissible four-resolution subsets containing $r = 1$, as the percentage of subsets with lower error (0 is best), and the column averages this percentage over the 60 cells. Rules that query other than four resolutions, and the oracles over other subsets, are ranked in the same per-cell distribution for reference, so their percentiles compare choices at the four-query budget rather than at their own.

Rule	$\mathrm{MSE}_{\mathrm{norm}}$	Gain	Queries	Percentile
Base, $r = 1$ only	0.5861	0.0%	1	95
Coverage rule of Appendix B	0.5613	4.2%	4	79
Uniform divisor-index spacing	0.5406	7.8%	4	58
Random subsets, five seeds	0.5407	7.7%	4	54
Fixed ladder $\{1, 2, 4, 8\}$	0.5303	9.5%	4	36
Best per-channel subset on CAL	0.5136	12.4%	4	15
Greedy shared-subset selection on CAL	0.5133	12.4%	4	13
All admissible resolutions	0.5052	13.8%	8–14	16
Per-channel oracle over four-resolution subsets	0.5061	13.6%	4	0

The budget counts backbone calls. Runtime also depends on the model and output length (Appendix D.2). Coverage selection needs eight queries to reach a gain comparable to the ladder with at most five. This differs from querying all admissible resolutions: for $P \geq 48$, the candidate set contains more than eight resolutions, and the all-admissible rule uses a mean of eleven calls.

The returns accumulate unevenly: for the ladder, moving from two resolutions to four increases the gain by 3.9 percentage points, while adding a fifth increases it by 0.5 points. The fifth rung extends the geometric ladder by the next power of two, 16, and any rung that does not divide P is dropped, so at $P = 24$ the fifth entry coincides with the fourth.

Both selection rules are insensitive to regularization on this grid: varying η from 0 to 10^{-1} changes pooled gain by less than 0.05 percentage points. For the fixed ladder, increasing η to 1 reduces gain by 0.4 points. We use $\eta = 10^{-2}$ throughout.

For the CAL-selected resolution variant, we compare three allocations on the same 60 cells: (i) both selection and precision estimation on all CAL blocks, (ii) selection on even-indexed blocks and precision on odd-indexed blocks, and (iii) selection on even-indexed blocks with precision on all blocks. Their gains are 12.4%, 12.1%, and 12.2%, respectively. These results show little change

Table 9: Marginal returns of the query budget, pooled over the 60 cells. Budget is the maximum number of resolutions a rule may query, matching the at-most-K constraint of Section 3. At $K = 5$ the ladder uses four queries for $P = 24$, where the fifth rung 16 does not divide the horizon, and five queries for the other horizons, a mean of 4.75. The coverage rule never returns fewer than its K. The two rules spend the budget differently: the coverage rule spreads it over the resolution axis, the ladder takes the geometric ladder.

Rule	Budget	Gain
Coverage	$K = 1$	0.0%
	$K = 2$	0.6%
	$K = 3$	2.5%
	$K = 4$	4.2%
	$K = 6$	7.5%
	$K = 8$	10.3%
Fixed ladder	$K = 2$	5.6%
	$K = 3$	8.3%
	$K = 4$	9.5%
	$K = 5$	10.0%

in TEST error when selection and precision estimation use disjoint calibration blocks. CAL-based selection requires calibration forecasts for the candidate resolutions before TEST.

D.1 The Long-Context Base

The long-context baseline queries the model once at the original resolution. We compare it with AMRI on 128 cells covering Chronos-2, Chronos-Bolt, TimesFM 2.5, and Moirai-2, each on all eight datasets and four horizons. The input length is 4096 observations for Chronos-2, TimesFM 2.5, and Moirai-2, and 2048 for Chronos-Bolt, its native limit.

On this subset, pooled $\mathrm{MSE}_{\mathrm{norm}}$ is 0.5518 for the 512-observation baseline, 0.4995 for the long-context baseline, and 0.5081 for AMRI. The long-context baseline has lower error than AMRI on each of the four backbone-level averages. AMRI's pooled error is 1.7% higher. Thus a longer original-resolution history is an effective alternative on this subset. The separate matched-history experiment gives a 4.4% AMRI improvement over the 512-observation baseline across the full 224-cell grid (Table 1).

D.2 Wall-Clock Latency

We measure TEST inference on ETTh1 at $P = 96$ using one RTX 3090. Each run covers the same 54 origins and seven channels: 378 forecasts for the 512-point base or one long-context query, and 1512 forecasts for the four-resolution ladder. Ladder queries read 512 aggregated points at each resolution. Long-context queries read 4096 original-resolution points, or 2048 for Chronos-Bolt. We synchronize CUDA, use two warm-ups and seven timed repeats, and report median model-inference time. Each row uses the same batch size for all three methods. Model loading, context preparation, and the one-time CAL replay are excluded. Therefore, the table compares TEST inference, not total deployment cost.

Table 10: Matched ETTh1 TEST inference times at $P = 96$ on one RTX 3090, in seconds for all 378 channel–origin forecasts (four times as many backbone calls for AMRI). All three methods in a row use batch size 8. Timings are medians of seven repeats after two warm-ups. The last column is the ratio of AMRI to long-context time.

Backbone	Long input	Base	AMRI	Long	AMRI/Long
Chronos-2	4096	0.941	3.936	2.445	1.61
Chronos-Bolt	2048	1.322	3.133	2.655	1.18
TimesFM 2.5	4096	4.288	17.086	11.168	1.53
Moirai-2	4096	1.226	3.033	4.170	0.73

At batch size 1, the corresponding AMRI/long-context ratios are 3.93, 2.41, 1.70, and 2.32, respectively. The ranking therefore changes for Moirai-2. Reconciliation after the model calls uses a precomputed $P \times n_{\mathcal{R}}$ matrix-vector product.

E Reconciliation Baseline Specifications

The reconciliation baselines and weighting variants are specified here. All consume exactly the cached native forecasts of the deployed ladder ($r \in \{1, 2, 4, 8\}$, expand span) and the same CAL origins as AMRI, per channel.

Heuristic-shrinkage full covariance. The full stacked error covariance S_{cal} is estimated from residuals at n_{cal} CAL origins, and shrunk toward its diagonal as $\widehat{\Sigma} = \delta \operatorname{diag}(\operatorname{diag} S_{\text{cal}}) + (1-\delta) S_{\text{cal}}$ with $\delta = \min(0.5,\, n_{\mathcal{R}}/n_{\text{cal}})$, the shrinkage growing when CAL is short. The precision $\widehat{\Lambda}$ is computed per channel as the Moore–Penrose pseudoinverse of $\widehat{\Sigma}$, which coincides with $\widehat{\Sigma}^{-1}$ whenever the shrunk matrix is nonsingular and guards the rank-deficient case. The recovery then solves the $P \times P$ normal equations $A^\top \widehat{\Lambda} A\, \tilde{\mathbf{y}}_t = A^\top \widehat{\Lambda} \hat{\mathbf{y}}_{\mathcal{R},t}$ by a direct linear solve. The system was nonsingular on the evaluated cells.

Second-moment reliability. The deployed $\widehat{\Lambda}$ uses the centered error variances s_j^2. This variant replaces them with uncentered second moments $m_j^2 = \frac{1}{n_{\text{cal}}} \sum_{t \in \text{CAL}} e_{t,j}^2$, where s_j^2 and $e_{t,j}$ are entry j of $\mathbf{s}^2$ and $\mathbf{e}_t$, respectively, and the sum runs over the n_{cal} calibration origins. It uses the ridge $\eta \bar{m}^2$, with $\bar{m}^2 = n_{\mathcal{R}}^{-1} \sum_{j=1}^{n_{\mathcal{R}}} m_j^2$, so a stable bias in a measurement reduces its precision rather than escaping the weighting.

Unconstrained ridge combination. The free-combination diagnostic fits, per channel, a linear map $W_{\text{ridge}} \in \mathbb{R}^{n_{\mathcal{R}} \times P}$ from the stacked native measurements to the fine grid, $\tilde{\mathbf{y}}_t = W_{\text{ridge}}^\top \hat{\mathbf{y}}_{\mathcal{R},t}$, by ridge regression on the CAL rows. The stacked entries enter at their native scale in the per-channel normalized space, with no per-column standardization, and the penalty is scaled by the row count. The strength is chosen by three-fold cross-validation on CAL over $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ (-46.2% pooled). The TEST-best strength over the same grid extended to 10^3, an oracle, still loses 33.7%.

F Adaptation Baseline Specifications

The four adaptation comparators in Table 3 are evaluated using the paper's cached queries, origins, and channels, so all four are scored on the pooled 224-cell grid. SMR^2, ELF, and ORCA consume feedback from completed TEST windows. The residual cascade uses CAL-fitted parameters and the observed context without TEST forecast-error feedback. The implementations and reproduction scripts are available in the anonymous repository.

Three further Table 3 rows are internal baselines defined here. **CtxRetrieval** mixes the frozen $r = 1$ forecast with the average of the three futures that follow the nearest stored SELECT windows (Euclidean nearest neighbours of the trailing 24-step query window), the mixing weight chosen on CAL. **DriftDebias** subtracts a horizon-specific mean forecast error per channel. The CAL variant fixes this vector before TEST. The online variant averages six completed origins, initialized from the last six CAL errors. The **gradient correction head** is a two-layer multilayer perceptron (MLP), $\text{Linear}(P, 128)$, GELU, $\text{Linear}(128, P)$, on top of the frozen forecast. It is pretrained on CAL rows (40 epochs, Adam at 10^{-3}, batch 256, up to 8192 subsampled rows) and then updated at test time: after each scored origin it takes five Adam steps at $3 \cdot 10^{-4}$ on the realized targets whose forecast windows have ended, strictly after the forecast for that origin was issued, with a 4096-row replay buffer. It is run on two backbones (Chronos-2, Moirai-2) over the eight datasets, hence its 64-cell coverage.

SMR^2 (Gao et al., 2026b) queries the backbone at strides $\{1, 2, 4, 6, 8\}$ (falling back to $\{1, 2, 4, 8\}$ where a stride is not cached), fits a ridge residual corrector per stride, channel, and horizon step on residual lags from the even-indexed CAL origins, and weights the corrected strides by their positive coefficients of determination R^2 on the odd-indexed half. At TEST, realized residuals of past origins feed the lag features online. We skip the corrector when its CAL $R^2 \leq 0$.

ELF (Lee et al., 2025) is reimplemented from the paper. ELF combines the frozen forecast with a Fourier-domain ridge forecaster fitted to observed input–target pairs, initially from SELECT and CAL. It uses frequency retention 0.9 and ridge strength 20, and updates its regression statistics as new targets become available. Exponential weighting combines the two forecasts using recent and cumulative prediction losses, with weight-update rate 0.5, window 5 update steps, and a five-block warm-up. In our evaluation, updates occur once per scored block rather than every 200 steps, and the weighter loss is MSE in SELECT-standardized units rather than mean absolute scaled error (MASE).

ORCA (Dai et al., 2026) uses the official implementation with its proposed configuration (Bay refiner, `xy` input, Boltzmann router). The refiner takes the observed context and the frozen forecast as inputs and produces a corrected forecast. The router weights the correction using past forecast errors. CAL pre-fills the replay buffer and trains the refiner before TEST. For each non-overlapping TEST window, ORCA first predicts using the preceding completed window's error, then updates its parameters after the current window's targets become available.

RGMR-inspired cascade adapts the coarse-to-fine residual refinement of RGMR (Gao et al., 2026a) to our cached native-resolution forecasts. It broadcasts each coarse forecast to the fine grid, fits per-level ridge residual predictors on CAL, and selects sigmoid-gate thresholds on CAL. Unlike RGMR, this adaptation does not use projected fine-grid queries or earlier TEST residuals as predictor features. Its result should not be read as a reproduction of RGMR.

G Shared versus Step-Specific Affine Calibration

An affine correction can use one slope and intercept shared across all horizon steps, or a separate pair at each step. Both are fitted on CAL. Table 3 uses the shared correction. Table 11 compares the two parameterizations.

Table 11: An affine correction fitted on CAL under the two parameterizations, pooled over the 224 cells of the grid. Median and the count are over the per-cell gain against base.

Fit	$\mathrm{MSE}_{\mathrm{norm}}$	Gain	Median	Cells below -50%
One slope for all steps	0.5692	$+1.8\%$	$+0.9\%$	0
One slope per step	0.7820	-34.9%	-8.4%	55

The forecast–target relationship can change between CAL and TEST. For example, on ETTh1 with TimesFM 3.0 at $P = 48$, channel 0, the correlation at steps 40–46 has magnitude at most 0.06 on CAL but ranges from 0.46 to 0.85 on TEST. The step-specific CAL slope is 0.05 at step 40, suppressing a forecast that is more correlated with the target on TEST. This channel has 148% higher MSE than the single-resolution forecast. The example illustrates how a step-specific fit can discard useful forecasts when their relationship with the target changes.

Sharing the correction across steps gives lower pooled error than fitting each step separately (Table 11). Across the 4424 channels, the median shared slope is 0.98.

H Input-Side Adaptation Baselines

These baselines modify the input context and require additional backbone queries. CtxNorm and EMA-Detrend each use one original-resolution query, while CtxEnsemble averages three original-resolution queries with different context lengths.

CtxNorm standardizes the context using the mean and standard deviation of its trailing 48 steps, forecasts, and inverts the transformation. Its normalized MSE differs from the single-resolution baseline by at most 0.03% in every cell, giving little change in accuracy under these settings.

EMA-Detrend subtracts an exponential moving average with span 12 and adds its final value back to the forecast. It increases pooled error for every backbone. The median cell-level gain is -43.3%. The transformation restores the last trend value but does not extrapolate the removed trend.

Table 12: Input-side adaptation baselines, pooled over the 224 cells of the grid. Median is over the per-cell gain against base, and Wins over AMRI is the share of cells on which the method beats the deployed configuration.

Method	$\mathrm{MSE}_{\mathrm{norm}}$	Gain	Median	Wins over AMRI
Single-resolution forecast	0.5798	0.0%	0.0%	17.4%
CtxNorm	0.5798	−0.0%	−0.0%	17.4%
EMA-Detrend	1.0503	−81.2%	−43.3%	6.2%
CtxEnsemble	0.6694	−15.5%	−10.1%	6.7%
AMRI (ours)	0.5222	+9.9%	+5.1%	

CtxEnsemble averages forecasts from context lengths 96, 192, and 288, all shorter than the baseline context of 512. This increases pooled MSE for every backbone, with error increases ranging from 6.8% to 20.6%.

I Per-Dataset Comparison against Trained Baselines

The trained baselines use three seeds and multivariate inputs. They receive 21 channels on weather and 321 on electricity, while the frozen models are evaluated on one and 21 channels, respectively. Both training allocations score the same forecast origins and channels as the frozen models. Under SELECT+CAL, eligible training windows come from the first 70% of the series, with the last 10% of these window origins held out for validation. SELECT-only fits on the first 30% and uses CAL for validation. Both allocations use Adam with learning rate 10^{-3} and batch size 32, for up to 50 epochs with early stopping after eight epochs without validation improvement.

Table 13: Pooled normalized MSE against trained baselines (lower is better), over the same 224 cells. Trained models use three seeds. The training allocation applies to DLinear, PatchTST, and iTransformer. AMRI uses the same frozen forecasts and CAL weights in both rows. Best values are bold.

Training allocation	DLinear	PatchTST	iTransformer	AMRI
SELECT (CAL validation)	0.6101	0.6287	0.5973	**0.5222**
SELECT+CAL	0.5379	**0.5204**	0.5387	0.5222

Table 13 summarizes the trained-baseline comparison across the full grid. Table 14 resolves the comparison by dataset, macro-averaged over the seven backbones. The base and AMRI columns are shared by the two panels, since neither depends on the supervised budget.

Table 14: Horizon-averaged normalized MSE by dataset, macro-averaged over seven backbones (lower is better). Base uses one original-resolution query. AMRI uses the fixed ladder. The two groups use the training allocations described above, with three seeds and the same evaluation origins and channels.

			SELECT and CAL			SELECT alone		
Dataset	Base	AMRI	DLinear	PatchTST	iTrans.	DLinear	PatchTST	iTrans.
ETTh1	0.9619	0.9059	0.9064	0.8885	0.9000	0.9750	1.3623	1.0609
ETTh2	0.6497	0.6081	0.7020	0.6455	0.6907	0.8295	0.7476	0.7249
ETTm1	0.9132	0.7255	0.7087	0.6546	0.7084	0.7974	0.8046	0.8617
ETTm2	0.4406	0.3961	0.4467	0.3880	0.4272	0.4507	0.4373	0.4503
exchange_rate	0.2251	0.2135	0.2301	0.2728	0.2952	0.5241	0.3037	0.3192
weather	0.9197	0.7909	0.7416	0.7305	0.7472	0.7393	0.8036	0.8109
electricity	0.2056	0.2080	0.2222	0.2436	0.2159	0.2144	0.2205	0.2116
traffic	0.3223	0.3296	0.3458	0.3395	0.3250	0.3505	0.3501	0.3391

With SELECT+CAL available for training and validation, AMRI has lower error than all three trained baselines on ETTh2, exchange_rate, and electricity. PatchTST leads these four methods on ETTh1, ETTm1, ETTm2, and weather, and iTransformer on traffic. Including the single-resolution baseline,

AMRI is best on ETTh2 and exchange_rate. The baseline leads on electricity and traffic under both allocations.

With SELECT-only training, AMRI has lower error than the three trained baselines on seven datasets. DLinear leads on weather. Including the single-resolution baseline, AMRI is best on the four ETT datasets and exchange_rate. The trained methods have higher pooled error under this allocation, while base and AMRI use the same forecasts in both comparisons.

J Online Reliability Updates

The main method fixes its CAL weights throughout TEST. We examine an online extension that updates reliability after forecast targets become available. This uses the same cached multi-resolution forecasts, expanded-history inputs, and 224 cells as the main experiment. It requires no additional backbone queries or parameter updates.

Update rules. At each TEST origin, we re-estimate the entrywise error variances in Equation (9), retaining $\eta = 10^{-2}$, and recompute the combination matrix in Equation (10). *Expanding* uses all CAL errors and errors from completed TEST forecasts. *Rolling* uses the most recent $B_{\text{win}} \in \{16, 64, 256\}$ valid completed forecast blocks per channel, or all available blocks if fewer than B_{win} exist. Each block spans P time steps. Since forecast windows do not overlap, earlier TEST targets are observed before the next prediction. Current and future targets never enter its weights. Resolutions and SELECT normalization remain fixed. We specified all three windows before running this comparison and report each one. Rolling can truncate CAL at the first TEST origin, whereas Expanding starts with the same weights as the main method.

Table 15: Online reliability updates with expanded history. (a) Pooled normalized MSE and improvement counts over 224 cells. (b) Dataset-level relative MSE reductions. All reductions compare with fixed CAL weights, with positive values indicating lower error. Cells receive equal weight within each aggregate.

(a) Overall results

Update rule	MSE_{norm}	Reduction	Improved cells
Fixed CAL (main method)	0.522201	0.000%	–
Expanding	0.518424	+0.723%	172/224
Rolling, $B_{\text{win}} = 16$	0.519222	+0.570%	129/224
Rolling, $B_{\text{win}} = 64$	0.518207	+0.765%	167/224
Rolling, $B_{\text{win}} = 256$	0.517937	+0.817%	174/224

(b) Reduction relative to fixed CAL weights

Dataset	Expanding	Rolling 16	Rolling 64	Rolling 256
ETTh1	+1.755%	+1.736%	+1.802%	+1.755%
ETTh2	+0.042%	+0.096%	+0.091%	+0.045%
ETTm1	+1.545%	+1.087%	+1.499%	+1.780%
ETTm2	+0.266%	+0.920%	+0.987%	+0.824%
exchange_rate	−0.150%	−0.107%	−0.149%	−0.150%
weather	+0.010%	−0.152%	−0.021%	+0.018%
electricity	+0.940%	+0.518%	+0.894%	+0.958%
traffic	+0.027%	−1.106%	−0.331%	−0.017%

Results. Online updates yield modest additional reductions in pooled MSE (Table 15). Expanding reduces error by 0.723% and improves 172 of 224 cells without a window-length choice. Rolling with $B_{\text{win}} = 256$ gives the lowest pooled error among the tested settings, a 0.817% reduction, but is not uniformly better: exchange_rate and traffic deteriorate slightly. Shorter windows also increase traffic error. These results suggest that AMRI can exploit feedback from completed forecasts, while the benefit depends on the dataset and calibration window. The online variants receive TEST error feedback unavailable to the fixed-weight method. Their gains are reported separately from the main results.