CALIBRATE FIRST, REMEMBER THE RESIDUAL: OUTCOME-ALIGNED MEMORY FOR FORECASTING

Anonymous authors
Anonymous manuscript

ABSTRACT

Online forecasting provides a stream of delayed targets that can be used to correct later predictions. A short residual memory, however, may spend much of its capacity tracking persistent changes in forecast level and scale. We introduce Calibrated-Residual Outcome-Aligned Ridge Memory (CR-OARM), which first applies causal channel-wise affine calibration and then fits a 32-episode ridge memory to the remaining error. The memory groups overlapping forecasts by their target timestamp so that repeated predictions of one outcome do not act as independent evidence. A frozen sample-wise gate controls correction strength. Across six datasets, four horizons, two backbones, and three source seeds, CR-OARM obtains the lowest mean MSE in a 144-case matched comparison (0.3826, versus 0.3865 for calibration, 0.3874 for COSA, and 0.3889 for TAFAS), although paired intervals against the two adapters include zero. In a convergence-controlled replay, a gate retrained on purged validation streams matches calibration (0.3859 versus 0.3858), while the fixed original gate reaches 0.3820 MSE and is reported as a control. A prediction-equivalent compact implementation runs 1.84–2.79$\times$ faster than released TAFAS on four GPU cases.

1 INTRODUCTION

After a seasonal change, an electricity forecaster may become consistently too low. Recent targets can reveal this bias, but a small adaptation memory must then represent both systematic drift and context-dependent error. The two signals compete for the same limited capacity.

Online residual correction keeps the source model fixed and learns from errors as targets arrive (Im & Kwon, 2026; Lee et al., 2025; Kim et al., 2025; Joulani et al., 2013). Regression-based test-time memory offers a lightweight way to store these errors (Sun et al., 2025; Wang et al., 2026b; Peng et al., 2026). Two details are easy to overlook. First, a memory fitted to raw source residuals must relearn level and scale shifts that a low-dimensional calibrator can already capture. Second, long-horizon forecasting produces overlapping episodes: several forecast origins can predict the same future timestamp. Treating those forecast instances as independent lets repeatedly predicted outcomes dominate the fit.

We address both issues with a calibrate-then-remember pipeline. A causal per-channel affine model tracks level and amplitude errors. Each calibrated forecast is stored when issued; once its targets arrive, only the remaining error enters memory. Outcome-Aligned Ridge Memory (OARM) fits these residuals with a group precision indexed by target timestamp, and a frozen sample-wise gate sets the correction strength.

Contributions. Our contributions are threefold. (1) We formulate a causal calibrate-then-remember adapter whose memory target excludes affine drift and whose group precision accounts for forecasts sharing a target timestamp. (2) We separate the roles of calibration, outcome alignment, and gating through matched adapter comparisons, ordering and covariance controls, convergence-controlled training, and controlled simulations. (3) We derive a compact implementation that avoids the expanded design matrix while preserving predictions, reducing end-to-end streaming time relative to released COSA and TAFAS implementations.

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.

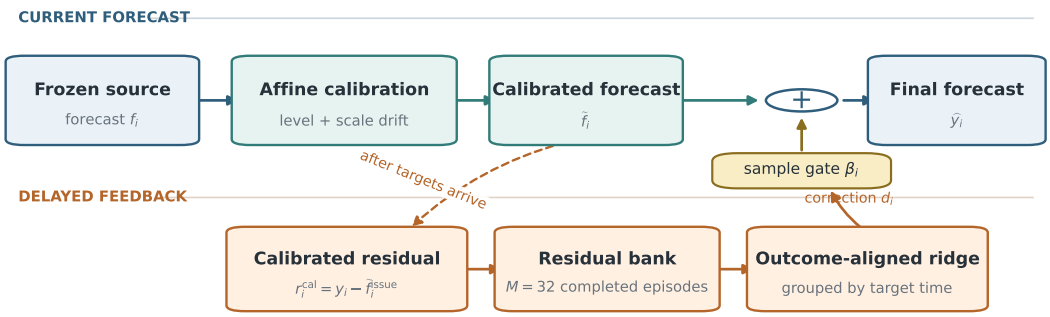

Figure 1: CR-OARM separates the current forecast path from delayed feedback. Calibration handles level and scale drift; after targets arrive, the residual bank fits an outcome-aligned correction that is scaled before being added to the calibrated forecast.

2 RELATED WORK

Memory as regression. Test-time training treats model state as a sequence-dependent memory (Sun et al., 2025), while test-time regression connects this view to associative recall (Wang et al., 2026b). Recent systems study learned memory states, retention rules, caching, and curvature-aware updates (Behrouz et al., 2025c;a;b; 2026; Tumma et al., 2026). Related linear-memory views include fading-memory ridge and Kalman updates (Peng et al., 2026), as well as local regression inside attention (Zuo et al., 2025; 2026). Our setting differs in the structure of the observations: feedback is delayed, and multiple stored forecast instances may refer to one realized target. We encode that dependence directly in the regression precision.

Online forecasting and retrieval. Online forecasters can combine source and online predictions (Lee et al., 2025), adapt directly in output space (Im & Kwon, 2026), or calibrate using progressively observed targets (Kim et al., 2025). Delayed-feedback methods emphasize when labels become available and which information an update may use (Huang et al., 2026; Liang et al., 2024; Wang et al., 2026a). Selective adaptation and retrieval instead decide when to correct or which historical examples to reuse (Zhang, 2026; Han et al., 2025; Tire et al., 2024; Zhou et al., 2026a;b; Villa Serna et al., 2026). OARM is complementary to these choices: it changes how a compact residual bank is fitted when retained forecasts share outcomes.

Calibration before memory. Calibration commonly adjusts forecasts before evaluation or online updating (Kim et al., 2025; Im & Kwon, 2026). Here it also defines the regression target. Memory stores the error left by an affine correction that was available when the forecast was issued, separating low-dimensional drift from context-dependent residual structure.

Statistical foundations. Recursive least squares and directional forgetting control how old information is retained (Goel et al., 2020; Zhu et al., 2022). Grouped deletion and time-aware validation address dependence during model selection (Golub et al., 1979; Pahikkala & Airola, 2016; Bergmeir et al., 2018), while forecast reconciliation uses structured covariance to couple related predictions (Di Fonzo & Girolimetto, 2023). We use an outcome-indexed working covariance inside a finite online correction memory. It downweights shared outcome error without forcing forecasts from different origins to agree.

3 CALIBRATE FIRST, THEN REMEMBER

Figure 1 shows the causal pipeline. At each decision cutoff, the method (i) calibrates the frozen source forecast, (ii) admits only fully observed episodes to a residual bank, (iii) fits an outcome-grouped ridge correction, and (iv) scales that correction with a frozen gate. This section defines the predictor; Section 4 distinguishes the gate variants used in evaluation.

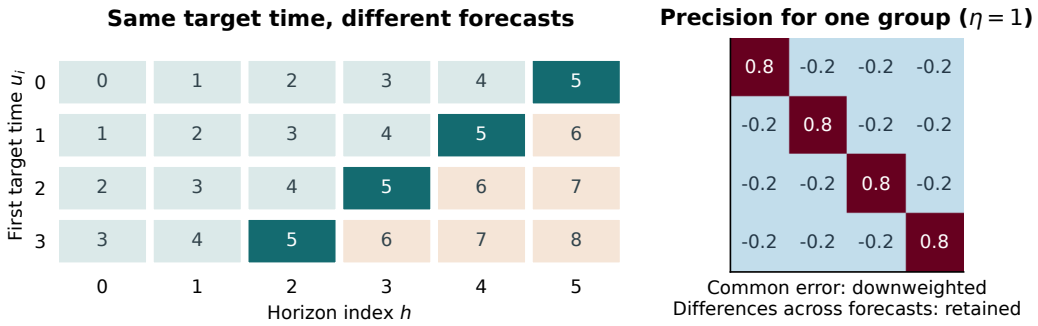

Figure 2: Each cell represents one forecast instance (an origin–horizon pair); its number is the target timestamp. Dark cells refer to one shared outcome. For four such instances and $\eta = 1$, the precision has diagonal entries 0.8 and off-diagonal entries -0.2. It reduces common error without imposing equal forecasts. Pale orange cells lie beyond the illustrative cutoff.

3.1 Forecasts, residuals, and available information

Let u_i be the timestamp of the first target of episode i, and let $h \in \{0, \dots, H-1\}$. A frozen source model produces $f_{i,h,c}$ for channel c. At decision cutoff t, an episode enters memory only if $u_i + H - 1 \le t$, so all its targets have arrived. The FIFO bank retains the latest $M = 32$ completed episodes; incomplete forecasts occupy additional pending storage. Newly issued predictions are scored only at timestamps strictly after t.

3.2 Causal affine calibration

For each channel, an exponentially weighted ridge fit predicts the source residual from a constant and the source value. At cutoff t it yields

$$\widetilde{f}_{i,h,c} = f_{i,h,c} + a_{t,c} + b_{t,c} f_{i,h,c}. \tag{1}$$

The coefficients start at zero. Their 2×2 sufficient statistics use only targets observed by t, with half-life 1536 and ridge penalty .01 selected by validation (Section 4). Coefficients are shared across lead times and fitted separately per channel.

Overlapping forecast episodes may predict one timestamp repeatedly. Updating the calibrator with every copy would overweight that outcome. We therefore use each observed timestamp once, paired with the most recent forecast that was available before the target arrived. Pending memory stores the complete issued vector $\widetilde{f}_i^{\text{issue}}$. When its horizon matures, the memory target is

$$r^{\text{cal}}_{i,h,c} = y_{u_i+h,c} - \widetilde{f}^{\text{issue}}_{i,h,c}. \tag{2}$$

The issued forecast remains fixed when its residual is computed.

3.3 A small residual-memory model

We flatten the $96 \times C$ context and divide by $\sqrt{96C}$. A fixed Gaussian projection maps this key to 32 dimensions. Augmenting it with a constant gives $\phi_i \in \mathbb{R}^{33}$. An orthonormalized piecewise-linear horizon basis gives $\psi_h \in \mathbb{R}^8$. The correction for channel c is

$$d_{i,h,c} = \phi_i^\top W_c \psi_h = z_{i,h}^\top \theta_c, \qquad z_{i,h} = \phi_i \otimes \psi_h. \tag{3}$$

The ordering of the Kronecker product places eight horizon-intercept coefficients first. These are unpenalized; the other 256 coefficients receive ridge regularization. Let $Z \in \mathbb{R}^{nH \times 264}$ contain the design vectors $z_{i,h}$, $Y \in \mathbb{R}^{nH \times C}$ the calibrated residuals from Equation 2, and $D = \operatorname{diag}(0_8, 1_{256})$. Projection and horizon sharing keep the fit small. Basis ridge uses the same design with an independent-instance loss.

3.4 GROUP PRECISION AND FITTING

Each origin–horizon pair defines a *forecast instance*. Instances with the same target timestamp u_i+h form a group g. Within each channel, consider the working residual model

$$Y_g = Z_g\Theta + \mathbf{1}a_g^\top + E_g, \qquad V_g = I + \eta\mathbf{1}\mathbf{1}^\top. \tag{4}$$

Here $\eta \geq 0$ is the ratio between common and independent noise variance after fixing the independent scale to one. The exchangeable block serves as a working covariance model; forecast origins may still differ in conditional bias and exposure to disturbances. We test its contribution against count weighting, degree-of-freedom matching, and shuffled timestamp groups.

With block-diagonal $P_\eta = \mathrm{blockdiag}(V_g^{-1})$, the fit and its candidate correction are

$$\widehat{\Theta}_\eta = (Z^\top P_\eta Z + n\lambda D)^{-1} Z^\top P_\eta Y, \qquad d_{i,h,c} = z_{i,h}^\top \widehat{\theta}_{\eta,c}. \tag{5}$$

We solve the linear system directly. The factor n counts episodes. All controls use this convention except the explicitly DF-matched penalties. Global-blend controls use $\widehat{y} = f + \beta d$ with validation-selected $\beta \in \{0, .25, .5, 1\}$; the pre-calibration memory baseline uses its sample-wise gate.

For a group with m_g forecast instances, the Sherman–Morrison identity gives

$$V_g^{-1} = I - \frac{\eta}{1+\eta m_g}\mathbf{1}\mathbf{1}^\top. \tag{6}$$

For any group residual vector e_g, write $e_g = \bar{e}_g\mathbf{1} + e_g^\perp$, with $\mathbf{1}^\top e_g^\perp = 0$. Its contribution becomes

$$e_g^\top V_g^{-1} e_g = \|e_g^\perp\|^2 + \frac{m_g}{1+\eta m_g}\bar{e}_g^2. \tag{7}$$

The precision attenuates the common direction; inverse-count weighting, $P_g = I/m_g$, attenuates both terms.

Within a timestamp group, residual differences cancel the shared realized target. The disagreement term in Equation 7 therefore softly penalizes differences between the *corrected source forecasts*. Raising η weakens the shared target anchor relative to this penalty. Excessive η can favor agreement between forecasts made with different information. Appendix F derives this relation.

The system is fitted after four completed episodes and refitted every eight new admissions. Coefficients remain fixed between refits; group precision is recomputed from the updated bank. Before the first fit, CR-OARM uses calibration alone, while ungated controls use the mean completed residual.

Compact deployment. The deployed predictor needs one outcome-aligned solve; the ordinary-ridge branch is an ablation. We store projected keys, cache unchanged bank summaries, and retain one latest-issued forecast for affine feedback. Separable feature contractions and target-group sums recover the same system without the expanded $nH \times 264$ design. Appendix K gives the identities and equivalence checks.

3.5 SAMPLE-ADAPTIVE CORRECTION STRENGTH

Correction reliability varies across forecast origins. CR-OARM predicts one scalar for each origin:

$$\beta_i = \sigma(g_\omega(s_i)), \qquad \widehat{y}_{i,h,c} = \widetilde{f}_{i,h,c} + \beta_i d_{i,h,c}^{\mathrm{cal}}, \tag{8}$$

where g_ω is a two-hidden-layer MLP with width 24 and GELU activations. Its 44-dimensional input s_i contains the projected current context, summaries of d_i^{cal} and $\widetilde{f}_i$, their cosine similarity, and five causal bank summaries. The gate sees neither y_i nor the current residual. The main 144-case comparison uses one gate trained at horizon 96 for the pre-calibration memory model and freezes its weights across all CR-OARM cases. We also train an architecture-identical gate on causal calibrated residuals from held-out validation streams. The two variants separate transfer of a fixed shrinkage policy from supervision matched to the calibrated residual target.

For a completed calibrated-residual example, the best bounded scalar for the fixed candidate is available in closed form:

$$\beta_i^* = \mathrm{clip}_{[0,1]} \frac{\langle y_i - \widetilde{f}_i, d_i^{\mathrm{cal}}\rangle}{\langle d_i^{\mathrm{cal}}, d_i^{\mathrm{cal}}\rangle + \epsilon}. \tag{9}$$

This quantity is unavailable at prediction time. The fixed original gate was trained analogously on the pre-calibration memory correction, with a smooth-L_1 loss plus normalized forecast regret

$$\mathcal{L}_i = \mathrm{Huber}(\beta_i, \beta_i^*) + \frac{2\beta_i\langle f_i - y_i, d_i\rangle + \beta_i^2\langle d_i, d_i\rangle}{\bar{R}_s}, \tag{10}$$

where $\bar{R}_s$ is the mean frozen error of the training stream. The denominator changes numerical scale, not the minimizer within a stream. The final gate is shared across datasets and source models, is frozen before evaluation, and is not updated on the test stream.

4 Experiments

4.1 Protocol and comparison

Benchmarks and source models. We evaluate ETTh1, ETTh2, ETTm1, ETTm2, Exchange, and Weather with DLinear (Zeng et al., 2023) and PatchTST (Nie et al., 2023). The ETT datasets follow Zhou et al. (2021). Every task uses 96 context points and a horizon in $\{96, 192, 336, 720\}$, giving 48 dataset–horizon–backbone combinations. The main comparison trains each source for 30 epochs from seeds 0, 1, and 2, evaluates validation MAE every five epochs, and retains the best checkpoint. All methods share that checkpoint within each of the resulting 144 cases. We later repeat the comparison after continuing every source to a common validation-plateau rule.

Streaming evaluation. The chronological train/validation/test ratios are 60/20/20 for ETT and 70/10/20 for Exchange and Weather, matching the released COSA loader. Channel standardization is fitted on training data, and validation/test windows include a 96-point historical prefix. For every batch, the observation cutoff is the final encoder timestamp; only targets strictly after that cutoff are scored. MSE and MAE average all forecast–horizon–channel instances. Validation and test streams both start with empty memory. Appendix G gives the exact indices and normalization.

Matched tuning. Each adapter family receives 12 validation candidates on every seed-0 task. Global-blend ridge variants cross $\lambda \in \{.01, .1, 1\}$ with $\beta \in \{0, .25, .5, 1\}$; COSA crosses feedback mode, learning rate, and update steps; TAFAS crosses feedback mode, learning rate, and weight decay. The selected setting is reused unchanged for seeds 1 and 2. Runtime is measured separately from this search.

Fixed CR-OARM settings. We fix $\eta = 1$ and $\lambda = .01$. Eight horizon-96 validation cases select the calibration half-life and penalty from $\{384, 1536\} \times \{.01, .1, 1, 10\}$. The selected pair, $(1536, .01)$, is then shared across datasets, horizons, and backbones. The original gate is trained for 30 epochs with AdamW (learning rate 10^{-3}, weight decay 10^{-4}) on pre-calibration memory examples at $H = 96$, then frozen for every test case. Because those residual features come from the same fitted source models, we distinguish this gate from the held-out and convergence-controlled variants below.

Held-out-gate control. For this control, calibrated residuals are generated sequentially on seed-0 validation streams. The first 60% trains three neural seeds and the final 40% selects one; the selected seed is then refit on all validation examples. No test target enters gate training or selection.

All reported scores include the startup period before memory reaches full capacity. Settings are fixed before evaluation and shared across source seeds. Appendix J describes numerical and temporal checks, and Table 4 isolates the pipeline components.

In error tables, **bold** and underlining mark the best and second-best values per metric, with ties sharing a style. Runtime and diagnostic quantities retain their own units.

4.2 Comparison with forecasting adapters

Tables 1–2 summarize the 144 matched test cases. Each dataset average contains 24 cases, and each horizon average contains 36.

Table 1: Main method comparison: test error across 144 matched cases. COSA/TAFAS use released update rules, observed-target feedback, and seed-0 validation-selected settings. Columns report MSE except the final MAE column.

Method	ETTh1	ETTh2	ETTm1	ETTm2	Exchange	Weather	Mean MSE	Mean MAE
No adaptation	**0.5700**	0.3284	0.5031	0.2429	0.4268	0.2876	0.3931	0.4127
COSA (matched)	0.5762	0.3274	0.4767	0.2427	0.4212	0.2803	0.3874	0.4104
TAFAS (matched)	0.5778	0.3230	**0.4764**	0.2408	0.4242	0.2911	0.3889	0.4123
Calibration only	0.5844	0.3266	0.5058	0.2319	**0.3979**	0.2724	0.3865	0.4112
CR-OARM (held-out gate)	0.5875	0.3239	0.5003	0.2314	0.4047	**0.2660**	0.3857	0.4121
CR-OARM (proposed)	0.5840	**0.3170**	0.4981	**0.2270**	0.4031	0.2662	**0.3826**	**0.4102**

CR-OARM records the lowest mean MSE for every source seed. Across all cases, it beats validation-selected COSA in 93/144 comparisons and TAFAS in 88/144. Its mean MAE is within 0.0003 of COSA.

Table 2: Test MSE by horizon over the same 144 cases. Adapter settings selected on seed 0 remain unchanged on seeds 1–2.

Method	Overall	96	192	336	720
CR-OARM (proposed)	**0.382564**	0.268397	**0.326406**	**0.391115**	**0.544336**
CR-OARM (held-out gate)	0.385652	0.269588	0.328698	0.394080	0.550241
Calibration only	0.386498	0.268371	0.327492	0.395171	0.554959
Ordinary ridge (same gate)	0.386475	0.271340	0.329501	0.395264	0.549793
Raw-residual memory	0.387395	0.270554	0.327659	0.393736	0.557630
COSA	0.387417	**0.266369**	0.329455	0.392400	0.561444
TAFAS	0.388881	0.266730	0.327751	0.395672	0.565370
No adaptation	0.393122	0.272358	0.330874	0.399688	0.569567

COSA leads at H96, whereas CR-OARM has the lowest MSE at H192, H336, and H720. By dataset, CR-OARM leads on ETTh2 and ETTm2; calibration leads on Exchange, the held-out gate on Weather, the frozen source on ETTh1, and TAFAS on ETTm1. Appendix D separates source seeds and backbones.

The gains vary across tasks. Task- and dataset-cluster bootstrap intervals for the mean difference from COSA and TAFAS include zero. Appendix D reports the intervals and seed-specific results; Figure 3 shows all seed-0 cases.

Comparison with published numbers. Appendix A reports the original COSA and TAFAS results separately (Im & Kwon, 2026; Kim et al., 2025). We do not merge those numbers into the matched ranking because the source checkpoints and scoring protocols differ. The appendix also records the difference between COSA's stated ETT split and its released loader.

Convergence-controlled confirmation. We repeat the full evaluation after continuing every source with fresh Adam state to a common stopping rule: at least 30 continuation epochs, no validation-MAE improvement above 10^{-4} for 20 epochs, and a learning rate at most one quarter of its initial value. All 144 sources satisfy the rule; 41 continue beyond 30 epochs and the longest run stops at epoch 114. After freezing these sources, we train three new gate seeds to a validation plateau. The first 60% of each horizon-96 validation stream supplies training examples, a 192-origin purge separates the segments, and the final 40% selects the gate. Test evaluation follows source and gate selection.

Table 3: Convergence-controlled confirmation on 144 test cases, split by source backbone. Each backbone column averages 72 cases. All methods use the same plateau-selected source checkpoints and future-only scoring masks. The original gate is evaluated without retraining; the retrained gate follows the purged validation protocol. Lower is better.

	DLinear		PatchTST		Overall	
Method	MSE	MAE	MSE	MAE	MSE	MAE
Frozen source	0.386165	0.405705	0.400590	0.418009	0.393378	0.411857
Calibration only	0.381332	0.405598	0.390236	0.414827	0.385784	0.410213
CR-OARM (retrained gate)	0.380580	0.407491	0.391126	0.416961	0.385853	0.412226
Ordinary ridge (same gate)	0.380967	0.407679	0.391461	0.417204	0.386214	0.412442
CR-OARM (fixed original gate)	0.376995	0.404650	**0.387078**	**0.414001**	**0.382036**	0.409326
COSA-P (released)	0.380987	0.404433	0.395726	0.416492	0.388357	0.410463
TAFAS (released)	**0.374593**	**0.399773**	0.390769	0.414183	0.382681	**0.406978**

The two backbones produce different rankings. For DLinear, TAFAS has the lowest MSE and MAE, while the fixed-original-gate control is second in MSE. For PatchTST, the fixed-original-gate control leads both metrics and reduces MSE from 0.390769 for TAFAS to 0.387078. Retraining the gate improves DLinear over calibration (0.380580 versus 0.381332 MSE), but not PatchTST (0.391126 versus 0.390236). Averaged over both backbones, the retrained gate and calibration are nearly identical (0.385853 and 0.385784 MSE), with a task-cluster interval of $[-0.00250, 0.00277]$. The fixed-original-gate control has the lowest overall MSE at 0.382036 and wins 93/144 cases against COSA-P and 78/144 against TAFAS. TAFAS has the lowest overall MAE.

4.3 Ablations and failure cases

Calibration order and memory components. Table 4 tests the order of calibration and memory. Calibrating first gives lower MSE than calibrating a memory-corrected forecast or adding a raw-residual correction after calibration. The second comparison keeps the gate fixed and changes only the residual target. Replacing outcome alignment with ordinary ridge causes a small loss; removing the gate causes a larger one.

Table 4: Ordering and mechanism ablations over the same 48 cases. "Wins" compares each method with calibration alone. The held-out gate controls for the source of supervision; the original gate remains fixed.

Method	MSE	MAE	Wins vs. calibration
CR-OARM (proposed)	**0.382618**	**0.410368**	33
Ordinary residual ridge, same gate	0.383335	0.410842	31
CR-OARM (held-out gate)	0.385820	0.412236	31
Ordinary ridge, held-out gate	0.386674	0.412968	26
Raw-memory correction after calibration	0.383132	0.411249	28
Memory then calibration	0.384682	0.411485	29
Calibration only	0.386684	0.411436	0
CR-OARM without gate	0.470376	0.461021	2
Residual ridge without gate	0.518018	0.484942	0

Gate architecture. The selected gate uses one output and current-state features. Across three neural seeds, adding a usefulness head, GRU, or attention does not reduce MSE. Usefulness AUC remains near 0.5, while continuous shrinkage from the current state is stable across these controls (Appendix C).

Source of gate supervision. With the held-out gate, outcome alignment improves mean MSE over ordinary residual ridge by 0.000823; both task- and dataset-cluster intervals exclude zero. Its

interval against calibration includes zero. The fixed original gate was trained with residuals from the fitted source models. Appendices C–D report the complete supervision-source comparison.

Additional controls. Appendices H–M examine timestamp shuffles, DF/count controls, LOO weighting, partial feedback, and the contributions of bias and context. These controls separate covariance modeling from correction strength.

4.4 Mechanism and computational cost

Controlled simulations show that outcome alignment helps under high overlap and shared noise, while independent noise can favor ordinary ridge. Appendix I reports all 864 trials and DF-matched controls.

Table 5 compares three CR implementations with released COSA-P/TAFAS configurations over three complete-stream repetitions.

Table 5: Controlled end-to-end streaming cost: median μs/origin / peak allocated MiB over three runs. Bold/underline: lowest/second-lowest per metric, including Frozen. DL/PT: DLinear/PatchTST; E2/W: ETTm2/Weather; suffix: horizon.

Method	DL–E2–96	PT–E2–96	DL–W–96	DL–E2–720
Frozen	**16.9 / 20.3**	**473.6 / 173.2**	**19.6 / 25.7**	**16.0 / 29.4**
CR reference	156.2 / 57.8	589.6 / 175.6	169.5 / 64.0	625.0 / 315.9
CR single branch	121.9 / 57.8	573.6 / 175.5	153.2 / 63.9	484.1 / 315.9
CR compact	<u>68.8</u> / 42.8	<u>511.2</u> / <u>175.4</u>	<u>78.4</u> / 65.0	<u>84.4</u> / <u>60.9</u>
COSA-P released	593.9 / <u>23.4</u>	1249.4 / 211.8	1443.6 / <u>34.6</u>	609.2 / 126.9
TAFAS released	174.4 / 27.0	1426.2 / 213.9	178.1 / 43.3	155.4 / 154.6

Timing includes inference, feedback, updates, and prediction. Appendix K specifies measurement boundaries and verifies prediction equivalence, with maximum MSE/MAE difference 2.99×10^{-11} for compact CR.

Compact execution is 1.15–7.40 times faster than the reference CR implementation, 2.44–18.41 times faster than released COSA-P, and 1.84–2.79 times faster than released TAFAS on these four cases. At DLinear–ETTm2 H720, peak allocation falls from 315.9 to 60.9 MiB (80.7%). Weather H96 instead rises from 64.0 to 65.0 MiB; the other H96 peaks fall. Adapter tensor state decreases by 8.8–21.9%.

5 Discussion and conclusion

CR-OARM separates two sources of online forecast error. Causal affine calibration follows channel-wise level and scale drift; the residual bank then models context-dependent error. Target-time precision adds a smaller, consistent improvement over ordinary ridge when the gate is trained on held-out calibrated residuals. The ablations and controlled simulations connect this gain to overlapping forecasts with shared outcome noise.

The matched 144-case comparison favors CR-OARM in mean MSE, while uncertainty intervals against COSA and TAFAS include zero. The convergence-controlled replay further shows that the result depends on the gate: retraining on purged validation streams matches calibration, whereas the fixed original gate obtains the best overall MSE and is therefore kept as a separate control. TAFAS obtains the best overall MAE. These results identify calibration as the reliable gain and outcome alignment as a smaller contribution whose value depends on the residual structure and correction policy.

The compact formulation preserves the fitted predictor and reduces streaming time relative to the released COSA and TAFAS implementations on four GPU cases. The evaluation covers six datasets,

two backbones, four horizons, three source seeds, and 144 convergence-controlled source replays. Broader claims about hardware cost and gate transfer require measurements beyond the single-GPU timing set studied here.

Reproducibility Statement

The implementation records candidate settings, validation scores, selected parameters, source checkpoints, dataset hashes, split indices, and convergence traces. Tables and figures are generated from these records. Numerical tests verify the linear solves and degree-of-freedom controls; temporal tests verify observation cutoffs and memory admission. Appendix G specifies the forecasting protocol, and Appendix J describes implementation checks and cost measurements. Code and checkpoints are not yet public.

AI use statement

Generative AI assisted brainstorming, mathematical derivations, experimental design and analysis, code development, and manuscript preparation. The authors take responsibility for the content and scientific claims.

References

Ali Behrouz, Zeman Li, Praneeth Kacham, Majid Daliri, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. ATLAS: Learning to optimally memorize the context at test time. *arXiv preprint arXiv:2505.23735*, 2025a. URL https://arxiv.org/abs/2505.23735.

Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. It's all connected: A journey through test-time memorization, attentional bias, retention, and online optimization. *arXiv preprint arXiv:2504.13173*, 2025b. URL https://arxiv.org/abs/2504.13173.

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2025c. URL https://arxiv.org/abs/2501.00663.

Ali Behrouz, Zeman Li, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. Memory caching: RNNs with growing memory. *arXiv preprint arXiv:2602.24281*, 2026. URL https://arxiv.org/abs/2602.24281.

Christoph Bergmeir, Rob J. Hyndman, and Bonsoo Koo. A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 2018. URL https://robjhyndman.com/publications/cv-time-series/.

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. URL https://arxiv.org/abs/2006.08570.

Ankit Goel, Adam L. Bruce, and Dennis S. Bernstein. Recursive least squares with variable-direction forgetting: Compensating for the loss of persistency. *IEEE Control Systems*, 2020. doi: 10.1109/MCS.2020.2990516. URL https://arxiv.org/abs/2003.03523.

Gene H. Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979. doi: 10.1080/00401706.1979.10489751. URL https://doi.org/10.1080/00401706.1979.10489751.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan O. Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. *arXiv preprint arXiv:2505.04163*, 2025. URL https://arxiv.org/abs/2505.04163.

Xiannan Huang, Shuhan Qiu, Jiayuan Du, and Chao Yang. Online time series prediction using feature adjustment. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=s4U2FWEMTU.

Jeonghwan Im and Hyuk-Yoon Kwon. COSA: Context-aware output-space adapter for test-time adaptation in time series forecasting. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=L7Z5wBMPrW.

Pooria Joulani, András Gy"orgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, 2013. URL https://arxiv.org/abs/1306.0686.

HyunGi Kim, Siwon Kim, Jisoo Mok, and Sungroh Yoon. Battling the non-stationarity in time series forecasting via test-time adaptation. In *AAAI Conference on Artificial Intelligence*, volume 39, pp. 17868–17876, 2025. URL https://ojs.aaai.org/index.php/AAAI/article/view/33965.

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

Daojun Liang, Haixia Zhang, Jing Wang, Dongfeng Yuan, and Minggao Zhang. Act now: A novel online forecasting framework for large-scale streaming data. *arXiv preprint arXiv:2412.00108*, 2024. URL https://arxiv.org/abs/2412.00108.

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. URL https://arxiv.org/abs/2211.14730.

Tapio Pahikkala and Antti Airola. RLScore: Regularized least-squares learners. *Journal of Machine Learning Research*, 17, 2016. URL https://www.jmlr.org/papers/volume17/16-470/16-470.pdf.

Liangzu Peng, Aditya Chattopadhyay, Luca Zancato, Elvis Nunez, Wei Xia, and Stefano Soatto. Gated KalmaNet: A fading memory layer through test-time ridge regression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2511.21016.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): RNNs with expressive hidden states. In *International Conference on Machine Learning*, volume 267 of *PMLR*, pp. 57503–57522, 2025. URL https://proceedings.mlr.press/v267/sun25h.html.

Kutay Tire, Ege Onur Taga, Muhammed Emrullah Ildiz, and Samet Oymak. Retrieval augmented time series forecasting. *arXiv preprint arXiv:2411.08249*, 2024. URL https://arxiv.org/abs/2411.08249.

Neehal Tumma, Noel Loo, and Daniela Rus. Preconditioned DeltaNet: Curvature-aware sequence modeling for linear recurrences. *arXiv preprint arXiv:2604.21100*, 2026. URL https://arxiv.org/abs/2604.21100.

Juan Pablo Villa Serna, Rohan Asthana, and Vasileios Belagiannis. Model-agnostic retrieval-augmented extended forecasting for time series. *arXiv preprint arXiv:2608.14054*, 2026. URL https://arxiv.org/abs/2608.14054.

Haochun Wang, Ruichen Xu, Georgios Kementzidis, Karen Cho, Sebastian Ramirez Villarreal, and Yuefan Deng. Towards principled test-time adaptation for time series forecasting. *arXiv preprint arXiv:2605.17250*, 2026a. URL https://arxiv.org/abs/2605.17250.

Ke Alexander Wang, Jiaxin Shi, and Emily B. Fox. Test-time regression: a unifying framework for designing sequence models with associative memory. *Journal of Machine Learning Research*, 27, 2026b. URL https://www.jmlr.org/beta/papers/v27/25-0903.html.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI Conference on Artificial Intelligence*, 2023. URL https://arxiv.org/abs/2205.13504.

Yuchen Zhang. SAFE-TS: Causal selective residual adaptation with a frozen fallback. Software repository, 2026. URL https://github.com/litlelight/SAFE-TS. Accessed September 9, 2026.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI Conference on Artificial Intelligence*, 2021. URL https://arxiv.org/abs/2012.07436.

Shiqiao Zhou, Holger Schöner, Zipeng Wu, Edouard Fouché, I. A. G. Wilson, and Shuo Wang. Stationarity-aware retrieval-augmented time series forecasting. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2026a. URL https://arxiv.org/abs/2606.04135.

Shiqiao Zhou, Zipeng Wu, Holger Schöner, Edouard Fouché, I. A. G. Wilson, and Shuo Wang. Semantics-enhanced retrieval-augmented time series forecasting. *arXiv preprint arXiv:2606.14941*, 2026b. URL https://arxiv.org/abs/2606.14941.

Kun Zhu, Chengpu Yu, and Yiming Wan. Recursive least squares identification with variable-direction forgetting via oblique projection decomposition. *IEEE/CAA Journal of Automatica Sinica*, 9(3):547–555, 2022. doi: 10.1109/JAS.2021.1004362. URL https://www.ieee-jas.net/article/doi/10.1109/JAS.2021.1004362.

Yifei Zuo, Yutong Yin, Zhichen Zeng, Ang Li, Banghua Zhu, and Zhaoran Wang. Local linear attention: An optimal interpolation of linear and softmax attention for test-time regression. *arXiv preprint arXiv:2510.01450*, 2025. URL https://arxiv.org/abs/2510.01450.

Yifei Zuo, Dhruv Pai, Zhichen Zeng, Alec Dewulf, Shuming Hu, and Zhaoran Wang. Parallax: Parameterized local linear attention for language modeling. *arXiv preprint arXiv:2605.29157*, 2026. URL https://arxiv.org/abs/2605.29157.

A PUBLISHED RESULTS AND PROTOCOL COMPARISON

This supplement contains the protocol, controls, derivations, and implementation details used to support the main claims. The accompanying *Full Experimental Record* retains literature transcriptions, alternative scoring partitions, precursor studies, and complete condition-level results under a separate label sequence. We transcribe reference values from COSA (Im & Kwon, 2026), Table 2 (p. 7), and TAFAS (Kim et al., 2025), Table 1 (p. 17872). Both blocks cover the six shared datasets, four horizons, and DLinear/PatchTST. COSA-F uses a fixed adaptation batch, whereas COSA-P uses PAAS. TAFAS values in the COSA block were measured by the COSA authors; the second block comes from the TAFAS paper. Each block retains its own frozen baseline.

Splits and checkpoints. COSA Section 4.1 states a 70/10/20 split for all datasets, while its released loader uses 60/20/20 for ETT and 70/10/20 for Exchange and Weather. TAFAS reports the latter ratios. We follow the released loader. COSA's repository documentation also lists fixed-month ETT counts that differ from both descriptions, so the exact indices behind the published ETT scores are unavailable. Adapters in our main tables share source checkpoints; literature means use the source models trained by the respective authors.

Forecasts included in the metric. Our primary score includes targets strictly after the common batch cutoff. The released COSA/TAFAS full-horizon tensor also contains observed prefixes. Published reference values retain their own scoring protocol and are not pooled with the primary comparison. Aggregate literature entries also supply no paired per-instance loss series for our significance calculations.

Table 6: Audited half-open outcome intervals in the released-code benchmark. Validation and test input arrays also include the preceding 96 observations as context. These indices, normalization statistics, and dataset hashes are shared across all source seeds.

Dataset	Observations	Train	Validation	Test
ETTh1	17420	$[0, 10452)$	$[10452, 13936)$	$[13936, 17420)$
ETTh2	17420	$[0, 10452)$	$[10452, 13936)$	$[13936, 17420)$
ETTm1	69680	$[0, 41808)$	$[41808, 55744)$	$[55744, 69680)$
ETTm2	69680	$[0, 41808)$	$[41808, 55744)$	$[55744, 69680)$
Exchange	7588	$[0, 5311)$	$[5311, 6071)$	$[6071, 7588)$
Weather	52696	$[0, 36887)$	$[36887, 42157)$	$[42157, 52696)$

Table 7: Reported COSA Table 2 results: MSE averaged over DLinear/PatchTST and four horizons. TAFAS/PETSA entries are evaluations by the COSA authors. Ranks apply within this source block.

Method	ETTh1	ETTh2	ETTm1	ETTm2	Exchange	Weather	Mean
No adaptation	0.5578	0.3147	0.4814	0.2285	0.3709	0.2646	0.3696
TAFAS	0.5453	0.3055	0.4568	0.2261	0.3616	0.2521	0.3579
PETSA	0.5436	0.3062	0.4599	0.2260	0.3615	0.2546	0.3587
COSA-F	0.5237	**0.2807**	**0.4278**	**0.2007**	**0.1945**	**0.2226**	**0.3083**
COSA-P	**0.5134**	0.2877	0.4364	0.2079	0.2347	0.2286	0.3181

Table 8: Reported original TAFAS Table 1 results, over the same six dataset names, two architecture names, and four horizons. Means derive from the original three-decimal entries.

Method	ETTh1	ETTh2	ETTm1	ETTm2	Exchange	Weather	Mean
No adaptation	0.5483	0.3139	0.4765	0.2228	0.3581	0.2642	0.3640
TAFAS	**0.5343**	**0.3049**	**0.4542**	**0.2195**	**0.3262**	**0.2503**	**0.3482**

B RELEASED CONFIGURATIONS AND SCORING PARTITIONS

The 48-case released-configuration comparison uses the source checkpoints and CR-OARM settings from the main benchmark. Both adapters start independently from seed 19 with learning rate .001, weight decay .0001, partial feedback, and prediction adjustment. COSA-P uses context length 10, three update steps, mature-episode feedback, and its released adaptive learning-rate schedule. TAFAS uses one update step and gate initialization .01.

Table 9 separates future targets, observed prefixes, and the full horizon. For each case, full-horizon MSE/MAE is the count-weighted average of the two partitions. Averaging case scores can yield different mixture weights across aggregate columns.

The checks in Appendix J also verify the partition identity. Frozen, calibration, and both CR-OARM scores reproduce within 10^{-7}. Ten validation-selected adapter configurations coincide with released settings: nine reproduce MSE/MAE within 10^{-7}; COSA on PatchTST–Weather H192 differs by 1.76×10^{-7} MSE and 1.42×10^{-6} MAE. Each evaluation retains its measured values.

Using 50,000 paired bootstrap draws with seed 20260918, task-level 95% intervals for CR-OARM minus comparator are $[-0.01323, 0.00120]$ against COSA-P and $[-0.00736, 0.00697]$ against TAFAS. Resampling the six datasets gives $[-0.01779, 0.00679]$ and $[-0.01304, 0.01453]$, respectively. All intervals span zero. This comparison uses one source seed.

Table 9: Scoring partitions for the 48-case released-configuration comparison. Each MSE/MAE column is averaged over the same 48 tasks. Observed-prefix scores describe targets already available at the batch decision cutoff.

Method	Future MSE	MAE	Observed MSE	MAE	Full MSE	MAE
No adaptation	0.393777	0.413499	0.207936	0.276978	0.372120	0.395464
Calibration only	0.386684	0.411436	**0.197800**	0.275643	0.364512	0.393596
COSA-P (released)	0.388407	0.411780	0.203336	**0.274092**	0.366770	0.393512
TAFAS (released)	**0.382488**	**0.408230**	0.205011	0.275639	0.361732	**0.390665**
CR-OARM (held-out gate)	0.385820	0.412236	0.198785	0.277979	0.363867	0.394563
CR-OARM (proposed)	0.382618	0.410368	0.198801	0.278296	**0.360949**	0.392889

The 48 DLinear and 48 PatchTST case-level entries, together with their full-horizon partitions, are retained in the Full Experimental Record, Section B. The aggregate table above contains the quantities used in the manuscript comparison.

C GATE ARCHITECTURE AND SUPERVISION-SOURCE CONTROLS

Table 10 evaluates gate architectures on the held-out final 40% of validation streams over three neural seeds. The mismatched-history control supplies another sample's history. Table 11 compares source-training features with calibrated residuals from source-held-out validation data; Section 4 specifies training and selection.

Table 10: Gate ablation at $H = 96$. MSE is mean $\pm$ standard deviation over three gate seeds; source checkpoints are fixed. "Wins" compares each run with fixed $\beta = .25$.

Gate	MSE	Wins vs. fixed/12	Usefulness AUC
One-head, current only	**0.313685** $\pm$ 0.000252	8.3	0.498
One-head + GRU-attention	0.314309 $\pm$ 0.000328	8.3	0.492
Two-head + GRU	0.314310 $\pm$ 0.000106	8.0	0.503
Two-head + GRU-attention	0.314429 $\pm$ 0.000309	7.7	0.488
Two-head, current only	0.314488 $\pm$ 0.000416	7.7	0.510
Mismatched history	0.314546 $\pm$ 0.000363	8.3	0.498

Table 11: Gate supervision-source control. Bold and underline rank each metric within a column. The 24-case columns cover source seeds 1–2 only.

Gate / correction	48 cases, seed 0		24 $H = 96$ cases, seeds 1–2	
	MSE	MAE	MSE	MAE
Calibration only	0.386684	<u>0.411436</u>	<u>0.269107</u>	**0.335600**
Proposed raw-residual gate	**0.382618**	**0.410368**	**0.268997**	<u>0.336427</u>
Held-out calibrated-residual gate	<u>0.385820</u>	0.412236	0.270280	0.337625
Ordinary ridge, held-out gate	0.386674	0.412968	–	–

Training and inference accounting. The gate has two width-24 hidden layers and 1,705 trainable parameters. Gate training reuses cached frozen forecasts and OARM corrections; it does not backpropagate through or retrain DLinear, PatchTST, or the ridge solve. At deployment it adds one MLP evaluation per forecast origin. OARM is fitted after four completed episodes and refitted after every eight newly admitted episodes. Before the first fit, the candidate correction is zero. The gate weights remain fixed throughout each test stream.

Oracle and failure concentration. The bounded per-sample oracle reaches 0.369292 macro MSE, versus 0.387781 for SA-OARM, measuring the information available in the candidate correction. The largest SA-OARM–COSA gap is 0.056709 on PatchTST–Exchange at horizon 336; the next four gaps are at most 0.025516. This case changes the aggregate ordering if removed, and all reported aggregates include it.

The held-out pre-calibration memory control reaches MSE 0.390604, compared with 0.393777 without adaptation and 0.392622 for fixed OARM. It improves on those controls in 34/48 and 33/48 cases, while trailing COSA in the aggregate. The raw-memory comparison and its full paired intervals are in the Full Experimental Record, Section Q.

D THREE-SEED MATCHED COMPARISON

Tables 12 and 13 separate the main comparison by source architecture.

Table 12: DLinear: test MSE by dataset and overall MSE/MAE, averaging four horizons and three source seeds.

Method	ETTh1	ETTh2	ETTm1	ETTm2	Exchange	Weather	Mean MSE	Mean MAE
No adaptation	**0.5710**	0.3247	0.5017	0.2357	0.3976	0.2968	0.3879	0.4077
COSA (matched)	<u>0.5738</u>	0.3244	<u>0.4709</u>	0.2354	0.3976	0.2879	<u>0.3817</u>	**0.4053**
TAFAS (matched)	0.5890	0.3235	**0.4668**	0.2322	0.3924	0.3134	0.3862	0.4092
Calibration only	0.5850	0.3248	0.5125	0.2262	**0.3746**	0.2770	0.3834	0.4075
CR-OARM (held-out gate)	0.5872	<u>0.3218</u>	0.5062	<u>0.2246</u>	0.3811	**0.2706**	0.3819	0.4084
CR-OARM (proposed)	0.5844	**0.3147**	0.5037	**0.2204**	<u>0.3797</u>	<u>0.2710</u>	**0.3790**	<u>0.4065</u>

Table 13: PatchTST: test MSE by dataset and overall MSE/MAE, averaging four horizons and three source seeds.

Method	ETTh1	ETTh2	ETTm1	ETTm2	Exchange	Weather	Mean MSE	Mean MAE
No adaptation	<u>0.5690</u>	0.3320	0.5044	0.2502	0.4560	0.2783	0.3983	0.4177
COSA (matched)	0.5787	0.3304	**0.4825**	0.2500	0.4449	0.2727	0.3932	0.4154
TAFAS (matched)	**0.5666**	<u>0.3226</u>	<u>0.4860</u>	0.2495	0.4559	0.2688	0.3915	0.4155
Calibration only	0.5838	0.3284	0.4992	<u>0.2376</u>	**0.4212**	0.2677	0.3896	<u>0.4150</u>
CR-OARM (held-out gate)	0.5879	0.3261	0.4945	0.2382	0.4284	**0.2613**	<u>0.3894</u>	0.4157
CR-OARM (proposed)	0.5836	**0.3193**	0.4925	**0.2335**	<u>0.4266</u>	<u>0.2614</u>	**0.3862**	**0.4139**

Table 14 separates results by seed; Table 1 gives the dataset means under the fixed-configuration protocol in Section 4. H96 checkpoints for seeds 1–2 are shared with the horizon-96 robustness experiment; that experiment's per-seed comparator tuning is excluded here.

Table 14: Test MSE by source seed, averaged over six datasets, four horizons, and two backbones. The All column averages 144 cases. Bold and underline mark the best and second-best result per column.

Method	Seed 0	Seed 1	Seed 2	All
CR-OARM (proposed)	**0.382618**	**0.383029**	**0.382044**	**0.382564**
CR-OARM (held-out gate)	<u>0.385820</u>	<u>0.386003</u>	<u>0.385132</u>	<u>0.385652</u>
Calibration only	0.386684	0.387004	0.385807	0.386498
Ordinary ridge (same gate)	0.386674	0.386776	0.385974	0.386475
Raw-residual memory	0.387781	0.387963	0.386440	0.387395
COSA	0.387753	0.388016	0.386482	0.387417
TAFAS	0.389263	0.389531	0.387849	0.388881
No adaptation	0.393777	0.393738	0.391849	0.393122

For paired uncertainty, we first average the CR-OARM-minus-comparator MSE over three seeds within each of the 48 dataset–backbone–horizon tasks, then resample tasks 50,000 times. The 95% intervals are $[-0.01236, 0.00185]$ against COSA and $[-0.01534, 0.00178]$ against TAFAS. Resampling the six datasets as clusters gives $[-0.01530, 0.00764]$ and $[-0.01861, 0.00726]$; task-level Wilcoxon p-values are .247 and .268. For the held-out gate, outcome alignment versus ordinary residual ridge gives a task-cluster interval of $[-0.00141, -0.00032]$ and a dataset-cluster interval of $[-0.00168, -0.00022]$.

Convergence-controlled confirmation details. All 144 source continuations satisfy the validation-plateau rule. Stop epochs range from 30 to 114 (median 30): 41 sources continue beyond epoch 30, 105 obtain a new best checkpoint, and 39 retain the original checkpoint. The three gate runs stop at epoch 30, with seed 29 selected by held-out validation MSE. Exchange contributes validation examples but no fitting examples because the purged first-60% segment contains no eligible origins. On DLinear, the fixed-original-gate control reaches 0.376995/0.404650 MSE/MAE, compared with 0.374593/0.399773 for TAFAS. On PatchTST it reaches 0.387078/0.414001, compared with 0.390769/0.414183 for TAFAS. For the retrained gate, task-cluster intervals are $[-0.01056, 0.00469]$ against COSA-P, $[-0.00492, 0.01022]$ against TAFAS, and $[-0.00250, 0.00277]$ against calibration. The corresponding dataset-cluster intervals are $[-0.01305, 0.00890]$, $[-0.00815, 0.01646]$, and $[-0.00533, 0.00670]$. All intervals use 50,000 paired bootstrap draws with seed 20260922.

E CALIBRATED-RESIDUAL RESULTS AND FAILURE PATTERNS

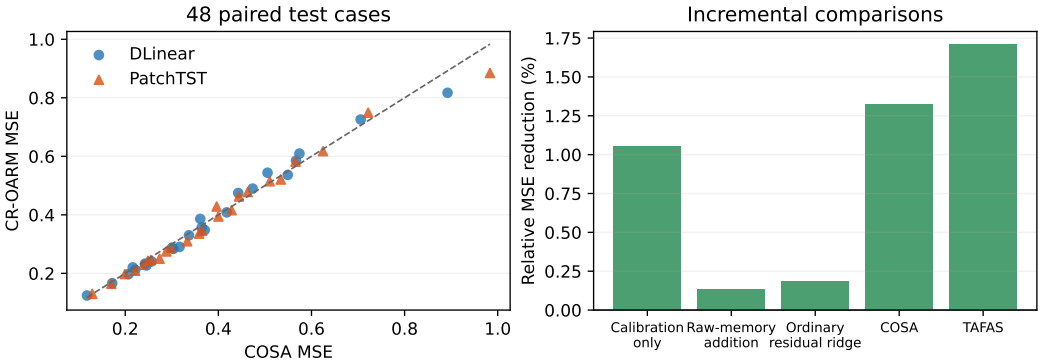

Figure 3: Seed-0 cases. Left: case-wise MSE against COSA; points below the diagonal favor CR-OARM. Right: relative MSE reduction against controls and comparators.

Table 15 summarizes all 48 seed-0 cases under the settings in Section 4. Complete case-level comparisons with validation-selected and released adapter settings, including every failure, are in the Full Experimental Record, Sections B and E.

Table 15: Seed-0 test MSE averaged uniformly over 48 cases. Horizon columns average 12 dataset–backbone tasks. "Wins" counts the lowest MSE among displayed methods. Bold and underline mark the best and second-best result per column.

Method	Overall	96	192	336	720	Wins/48
CR-OARM	**0.382618**	0.267196	**0.327067**	**0.395735**	**0.540474**	18
Raw-memory correction after calibration	<u>0.383132</u>	0.268034	<u>0.327325</u>	<u>0.396048</u>	0.541121	5
Calibrated residual ridge + gate	0.383335	0.268745	0.327628	0.396414	<u>0.540554</u>	3
Memory then calibration	0.384682	0.267853	0.327960	0.397842	0.545074	2
Calibration only	0.386684	0.266900	0.328400	0.399929	0.551507	7
Raw-residual memory	0.387781	0.269339	0.329127	0.400041	0.552617	4
COSA	0.387753	**0.265590**	0.331307	0.397879	0.556237	4
TAFAS	0.389263	<u>0.265607</u>	0.329310	0.401786	0.560348	12
No adaptation	0.393777	0.271046	0.332635	0.406509	0.564920	1

F DERIVATIONS AND ESTIMATOR SCOPE

F.1 THE ROLE OF SHARED-OUTCOME COVARIANCE

Linear-estimation identities clarify the outcome-alignment mechanism. In a scalar common-mean example, the information supplied by m instances with covariance $\tau^2 I + \sigma^2 \mathbf{1}\mathbf{1}^\top$ is

$$\mathbf{1}^\top V^{-1} \mathbf{1} = \frac{m}{\tau^2 + m\sigma^2}. \tag{11}$$

When $\sigma^2 > 0$, information saturates as additional noisy views share the same shock. This explains why repeated feedback can add less information than the number of instances suggests. Distinct forecast origins may still have distinct conditional means, and duplication can change the OARM fit.

A shrinkage-only special case. If every group contains one instance, $P_\eta = I/(1+\eta)$. Equation 5 then reduces to ordinary basis ridge with penalty $(1+\eta)n\lambda$. We match effective degrees of freedom (DF) to separate covariance geometry from this change in regularization. With $A_P = Z^\top P Z + n\lambda D$, the relevant quantity is $\mathrm{tr}(A_P^{-1} Z^\top P Z)$. This scalar matches overall model flexibility while leaving the estimators' geometry available for comparison.

Sensitivity to precision. For two precisions P and P_0 with the same design and penalty,

$$\widehat{\Theta}_P - \widehat{\Theta}_{P_0} = A_P^{-1} Z^\top (P - P_0)(Y - Z\widehat{\Theta}_{P_0}). \tag{12}$$

The correction changes only through residual directions affected by the precision difference and represented in the design. A timestamp relation that is statistically irrelevant can still change the estimator through this term. The derivation and conditional bias–variance expression follow below. Neither result guarantees an improvement with an estimated or misspecified covariance.

F.2 A FORWARD COVARIANCE SELECTOR ABLATION

Fixed $\eta = 1$ is the primary method. We also evaluate a separate OARM-forward variant. At each refit it reserves the newest four completed episodes as probes. An older episode is eligible for internal training only if its final target precedes the first probe's first target. If fewer than four eligible episodes remain, the selector uses $\eta = 0$. Otherwise it compares $\eta \in \{0, .25, 1, 4\}$ using unblended residual-prediction MSE on the probes, separately for each candidate λ. Exact ties prefer smaller η. It then refits the selected covariance on *all* current completed episodes.

This internal split uses feedback observed by the current decision. A temporal embargo separates training and probe outcomes. The selector requires additional fitting and search beyond the shared outer λ, β grid, and its compute budget is reported separately from fixed OARM.

Block precision. For $V = I + \eta \mathbf{1}\mathbf{1}^\top$, direct multiplication shows that $(I - \eta \mathbf{1}\mathbf{1}^\top/(1+\eta m))V = I$. Applying the inverse to $e = \bar{e}\mathbf{1} + e^\perp$ gives $V^{-1}e = \bar{e}\mathbf{1}/(1+\eta m) + e^\perp$, which proves Equation 7. A common error and a disagreement between forecasts are therefore not interchangeable in the objective.

Conditional bias and variance. For actual forecasting feedback, let F_g contain the stored source forecasts and $J_g = I - \mathbf{1}\mathbf{1}^\top/m_g$. Since $Y_g = \mathbf{1}y_g^\top - F_g$,

$$J_g(Y_g - Z_g\Theta) = -J_g(F_g + Z_g\Theta). \tag{13}$$

Thus the within-group disagreement term contains no new target information. Its useful role, if any, is a consistency bias between corrected forecasts, not additional independent supervision. The common component still contains the observed target. Forecasts from distinct information sets need not have equal conditional means, so reducing their target anchor can be harmful. This algebraic observation is separate from assuming exchangeable residual noise.

Consider a fixed design, a correctly specified conditional mean $Y = Z\Theta_* + E$, and independent output channels with common row covariance $\mathbb{E}[E_{:,c}E_{:,c}^\top \mid Z] = V$. Set $\alpha = n\lambda$, $A = Z^\top PZ + \alpha D$, and let $Q \in \mathbb{R}^{q\times p}$ be a fixed query design. With mean-zero errors and invertible A,

$$\widehat{\Theta}_P - \Theta_* = -\alpha A^{-1} D\Theta_* + A^{-1}Z^\top PE, \tag{14}$$

$$\frac{1}{qC}\mathbb{E}\|Q(\widehat{\Theta}_P - \Theta_*)\|_F^2 = \frac{\alpha^2}{qC}\|QA^{-1}D\Theta_*\|_F^2 + \frac{1}{q}\operatorname{tr}\left(QA^{-1}Z^\top PVPZA^{-1}Q^\top\right). \tag{15}$$

The first term is squared regularization bias and the second is variance; replacing P changes both. A correctly specified covariance can therefore increase error in a penalized prediction problem at a fixed nominal λ. If P is selected from Y, the fixed-P identity also omits selection risk. Mean misspecification, serial dependence beyond the timestamp blocks, and regime changes add further error. Classical unregularized GLS optimality consequently provides no OARM risk guarantee.

Precision sensitivity. The normal equation for $\widehat{\Theta}_{P_0}$ gives

$$A_P(\widehat{\Theta}_P - \widehat{\Theta}_{P_0}) = Z^\top PY - (Z^\top PZ + \alpha D)\widehat{\Theta}_{P_0} \tag{16}$$

$$= Z^\top (P - P_0)(Y - Z\widehat{\Theta}_{P_0}), \tag{17}$$

which proves Equation 12. Consequently,

$$\|Q(\widehat{\Theta}_P - \widehat{\Theta}_{P_0})\|_F \le \|QA_P^{-1}Z^\top\|_2 \, \|(P - P_0)(Y - Z\widehat{\Theta}_{P_0})\|_F. \tag{18}$$

This bound identifies the residual directions affected by a precision change. Because every term depends on the current fitted system, its magnitude alone does not select the estimator or predict the sign of the next-forecast gain.

Degrees of freedom. Partition $G = Z^\top PZ$ into the eight free intercept coordinates and the penalized coordinates: $G = \left[\begin{smallmatrix} G_{00} & G_{01} \\ G_{10} & G_{11} \end{smallmatrix}\right]$. The Schur complement is $S = G_{11} - G_{10}G_{00}^{-1}G_{01}$. If s_j are its nonnegative eigenvalues, then

$$\mathrm{df}(P, \alpha) = 8 + \sum_j \frac{s_j}{s_j + \alpha}. \tag{19}$$

The implementation matches this quantity to the fixed $\eta = 1$ fit by geometric bisection on the comparator's penalty. It uses the design and group labels only. DF-matched count weighting retains inverse-count weights without mean-one renormalization. DF matching controls overall flexibility; bias and risk can still differ because of the weighting geometry.

Grouped LOO and temporal dependence. With a fixed whitened design, covariance, and penalty, deleted-group ridge residuals satisfy $e_g^{(-g)} = (I - H_{gg})^{-1}e_g$. Efficient ridge and grouped-deletion computations are established (Golub et al., 1979; Pahikkala & Airola, 2016). Removing forecast instances targeting one timestamp can still leave that value in the input of a later context. The forward selector therefore uses a temporal embargo to approximate the information available at an earlier decision. It remains a small-sample model-selection heuristic.

G Experimental details

The main 144-case experiment uses the partitions in Table 6, the shared source architectures below, and the fixed-30-epoch selection rule in Section 4. Its source checkpoints, scoring masks, and seed-0 adapter selections are shared within every matched comparison.

Streaming protocol. For a PAAS batch starting at window index s with size b, the common observation cutoff is $t = s+96+b-2$. Window j has target timestamp $s+96+j+h$, for $j \in \{0, \ldots, b-1\}$ and $h \in \{0, \ldots, 95\}$. We score it only if $s+96+j+h > t$. Batch length is determined from the first context's dominant FFT period plus one, following the local PAAS implementation. This protocol allows adaptation to all observations available at a shared batch decision time; it is not per-origin immediate issuance.

The last window of each batch becomes a pending memory episode. For CR-OARM, this pending record contains the calibrated forecast produced at issuance. It is admitted only when all outcomes have arrived, and its residual is formed against that immutable record. Admission occurs before fitting at the current cutoff; the just-queued forecast cannot provide its own future targets. All memory-regression variants use the same FIFO episode origins and refit schedule. All comparators share masks and source predictions, but COSA retains its own batch-level update and buffer rules rather than the 32-episode bank. The score denominator counts every eligible forecast–horizon–channel cell, so a realized timestamp may contribute more than once. Outcome grouping is a fitting choice, not a change to the evaluation denominator.

Source architectures and training. The source architectures follow the released COSA implementation. DLinear uses a 25-point moving average and channel-shared linear heads. PatchTST uses patch length 16, stride 8, four encoder layers, eight attention heads, model and feedforward dimensions of 512, and dropout 0.1. Its built-in context normalization remains enabled in addition to the shared input normalization. Each checkpoint stores the complete configuration.

The source optimizer is Adam with batch size 128, initial learning rate 10^{-4}, and weight decay 10^{-4}; FP32 microbatches of 64 are accumulated to reach the full batch. The convergence replay uses ReduceLROnPlateau and the stopping rule in Section 4.

The Full Experimental Record, Sections F–G, contains the earlier four-case source comparison and complete definitions of full ridge, basis ridge, count/DF controls, LOO-ridge, and LOO-SVR.

H UNGATED-MEMORY CONTROLS

The four-case precursor study compared full and basis ridge, OARM, count/DF controls, timestamp shuffles, LOO weighting, SVR, COSA, and TAFAS under common forecast masks. On the two ETTh2 cases where the memory was active, fixed OARM improved basis-ridge MSE from 0.237344 to 0.236563 for DLinear and from 0.251260 to 0.249859 for PatchTST. At common $\lambda = .01, \beta = .25$, OARM also beat all five timestamp shuffles in every ETTh1/ETTh2 backbone case. The ETTh1 source remained stronger than every memory correction. Complete MSE/MAE tables, startup partitions, block-resampling intervals, and solver controls are retained in the Full Experimental Record, Sections G–H.

I SYNTHETIC GENERATOR AND COMPLETE MECHANISM RESULTS

I.1 CONTROLLED SHARED-NOISE EXPERIMENTS

We isolate conditional residual regression with 32 training episodes, 24 horizons, eight Gaussian context features, four horizon basis functions, three outputs, and 64 evaluation episodes. The grid crosses shared-noise fraction $\rho \in \{0, .5, .9\}$, origin stride $\{1, 6, 24\}$, and stationary, heteroscedastic, or coefficient-drift conditions. Thirty-two paired seeds give 864 trials. All methods use fixed $\lambda = .1$ and $\beta = 1$.

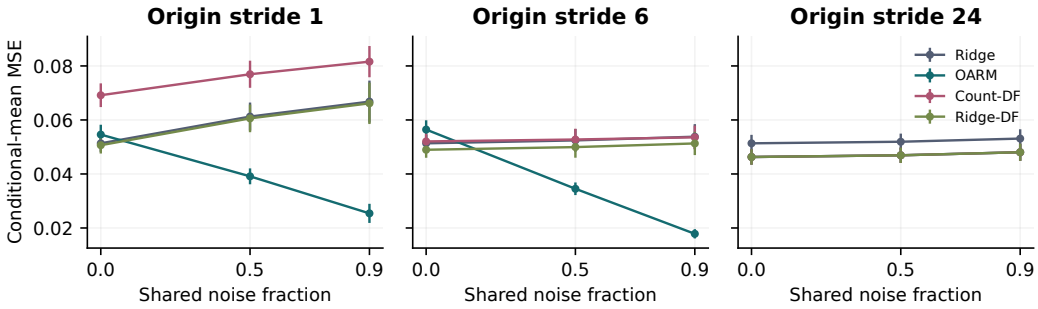

Figure 4: Conditional-mean prediction error in the stationary synthetic experiment. Bars are pointwise approximate 95% Monte Carlo intervals over 32 seeds. OARM benefits from shared noise at high overlap, but does not dominate when noise is independent. With stride 24, there is no overlap and OARM coincides with DF-matched ridge. Changing stride also changes the number of unique outcomes.

At stride 1, OARM changes conditional-mean MSE from 0.051349 to 0.054589 when $\rho = 0$, from 0.061251 to 0.039124 when $\rho = .5$, and from 0.066786 to 0.025412 when $\rho = .9$. The same pattern persists under heteroscedasticity and coefficient drift: shared noise favors outcome alignment, independent noise favors ordinary ridge, and no-overlap settings coincide with DF-matched ridge. The complete 27-condition table, generator equations, noisy-target metrics, and known-covariance control are retained in the Full Experimental Record, Section I.

J IMPLEMENTATION, VERIFICATION, AND COMPUTATION

The dense correction design has $N = 32H$ forecast instances and $p = (32 + 1) \cdot 8 = 264$ columns. At $H = 96$, $N = 3072$. One dense Gram/right-hand-side construction costs approximately $Np^2 + NpC$ multiply–accumulates (MACs): 219.8M for seven channels and 231.1M for Weather's 21 channels. The reference implementation constructs both ordinary-ridge and outcome-aligned branches; deployment uses the latter. A general LU solve costs approximately $2p^3/3 + 2p^2C$ floating-point operations (13.2M and 15.2M, respectively). These analytic counts exclude indexing, precision application, source inference, and the gate. One MAC represents two floating-point operations. The compact implementation in Appendix K changes system assembly while preserving the FP64 solve and estimator.

Solves use CUDA FP64 on an NVIDIA GeForce RTX 5060 Laptop GPU, with TF32 disabled and four CPU threads. System residual checks use $\|A\widehat{\Theta} - B\|_\infty / \max(1, \|B\|_\infty) < 10^{-8}$, with the infinity symbol denoting the maximum absolute entry. DF matching is checked to 10^{-7}. Unit tests compare group precision to direct inversion, verify the $\eta = 0$ reduction and orthonormal basis, check row-permutation equivariance and the no-overlap shrinkage identity, and test the temporal embargo. Additional checks recompute aggregate scores, verify that episode targets precede each cutoff, confirm fixed evaluation settings, and reproduce the full-ridge and frozen-source reference predictions.

Table 5 compares isolated execution with the released COSA-P/TAFAS configurations. A separate timing run with validation-selected settings and the two-branch reference implementation is retained in the Full Experimental Record, Section J. Controlled timing covers four cases on one GPU.

K Exact deployment reductions and isolated baselines

The implementation changes preserve the estimator, FP64 precision, FIFO capacity, random projection, horizon basis, calibration, gate weights, and admission/refit times. The single-branch variant removes the unused ordinary-ridge solve and prediction. The compact variant additionally changes how the same information is represented and contracted.

Sufficient-statistic assembly. Write Φ for the $n \times 33$ augmented projected keys and Ψ for the $H \times 8$ horizon basis. The expanded design is $Z = \Phi \otimes \Psi$. For target group g, let $s_g = \sum_{(i,h)\in g} z_{i,h}$, $v_g = \sum_{(i,h)\in g} r^{\mathrm{cal}}_{i,h}$, and $\alpha_g = (1 + m_g)^{-1}$, since $\eta = 1$. Equation 5 uses

$$Z^\top P Z = (\Phi^\top \Phi) \otimes (\Psi^\top \Psi) - \sum_g \alpha_g s_g s_g^\top, \tag{20}$$

$$Z^\top P Y = Z^\top Y - \sum_g \alpha_g s_g v_g^\top. \tag{21}$$

The unweighted right-hand side contracts each residual with Ψ, then contracts across episodes with Φ. Group sums are assembled from timestamp offsets and basis factors. This avoids materializing Z and its precision-transformed copy. Prediction similarly contracts each query key with the reshaped coefficient tensor before applying Ψ. The ridge penalty remains $n\lambda D$.

If T is the span of target timestamps retained by the bank, the leading group-Gram product is Tp^2 MACs instead of nHp^2. Constructing its group features adds order $nT \cdot 33 \cdot 8$ MACs. Base Gram factors, residual contractions and the TpC group right-hand side remain. Timestamp gaps have zero group sums. Savings therefore depend on overlap and horizon; these are implementation counts rather than measured hardware FLOPs.

Compact state. Each bank key is stored as its fixed 32-dimensional projection. The regressor and gate already consume only that projection. Target times follow from the stored origin and horizon, removing repeated index tensors. Five gate summaries are cached until bank membership changes. Affine feedback retains one latest-issued forecast: the final batch member starts at cutoff $+1$ and covers every still-pending timestamp of the preceding equal-length horizon. The old per-timestamp dictionary stored two identical source values; one forecast tensor and a release cursor preserve its latest-issued rule. Completed residual episodes remain distinct, retaining the disagreement information used by the precision and gate.

Accuracy and causal equivalence. Fourteen CUDA tests cover horizons 96, 192, 336 and 720; densely overlapping, separated and gapped origins; coefficient and prediction equivalence; immutable forecasts; feedback availability; FIFO eviction; and gate features. FP64 coefficients/predictions satisfy relative tolerance 10^{-8} and absolute tolerance 10^{-9}. Bank residuals and gate features match exactly. The four full-stream compact evaluations have maximum MSE/MAE difference 2.99×10^{-11} from the reference predictor. Independent COSA-P and TAFAS evaluations, each with independently initialized source and adapter states, reproduce their eight previously measured future-only metric pairs exactly (Table 16).

Table 16: Repeated accuracy evaluation on the four timing cases. The final column is the maximum absolute MSE/MAE difference from the 48-case evaluation. Scores are future-only.

Case	Method	MSE	MAE	Max score difference
DL-E2-96	CR compact	**0.166403**	0.275788	2.78e-17
DL-E2-96	COSA-P released	0.172729	0.276867	0.00e+00
DL-E2-96	TAFAS released	0.171183	**0.275440**	0.00e+00
PT-E2-96	CR compact	**0.164434**	**0.273910**	2.99e-11
PT-E2-96	COSA-P released	0.170786	0.275609	0.00e+00
PT-E2-96	TAFAS released	0.170207	0.275515	0.00e+00
DL-W-96	CR compact	**0.211698**	0.263567	5.55e-17
DL-W-96	COSA-P released	0.221378	0.264402	0.00e+00
DL-W-96	TAFAS released	0.221117	**0.262983**	0.00e+00
DL-E2-720	CR compact	**0.290435**	**0.363174**	5.55e-17
DL-E2-720	COSA-P released	0.326297	0.380803	0.00e+00
DL-E2-720	TAFAS released	0.309309	0.368452	0.00e+00

Controlled measurement. Each timing uses a fresh source and adapter state with two untimed source-forward warmups. Methods run sequentially; three repetitions rotate their order. Loading, initialization, metric scoring and report generation are excluded. Source inference, feedback processing, online updates/refits and prediction are included. CUDA is synchronized at both endpoints. Peak allocation includes the resident source and test tensor. State MiB counts unique adapter tensor storage, excluding the active frozen source and input table but including optimizer state, pending feedback and reset snapshots retained by the released implementations. It excludes Python-object overhead and host RAM. The per-run minima and maxima, together with all state-memory measurements, are retained in the Full Experimental Record, Section J; Table 5 reports the quantities needed for the accuracy–cost comparison.

L Additional controls and negative results

These controls use the four validation-stopped ETTh1/ETTh2 sources and remain separate from the 144-case benchmark. The Full Experimental Record, Sections L–O, contains every candidate, memory-fill subset, and numerical check.

Partial feedback and target anchoring. Partial OARM admits observed cells from pending episodes and removes them when the full episode matures. Its validation MSE is 0.324304, between fixed-schedule OARM at 0.323814 and basis-partial at 0.325020. A target-anchor precision that retains group-mean supervision reaches 0.324617, compared with 0.323814 for OARM, 0.324147 for DF-matched count weighting, and 0.324209 for count weighting. Neither extension improves the fixed OARM result. On ETTh1, the constrained global-blend optimum is $\beta = 0$ for both backbones and all three penalties.

Bias and context. Writing the affine correction as an anchor plus a zero-bank-mean contextual term separates moving bias from context. With each family's selected parameters, OARM context lowers FIFO32-bias MSE by 0.41% and 0.43% on the two ETTh2 cases and also lowers MAE. Recent-eight bias retains the MSE gains but increases MAE. At common $\lambda = .1, \beta = .5$, context improves every matched no-context control, while the complete ETTh1 forecasts remain worse than the source.

Online selection after bias correction. A causal ridge selector predicts the useful scale of each 24-step contextual correction from eight issuance-time features and 32 mature bands. Raw-unit

soft selection raises MSE and MAE in every active case: macro MSE is 0.325151 for full ridge and 0.324924 for OARM, compared with 0.324884 and 0.324581 for always-memory. Historical-scalar selection reaches 0.325104 and 0.324790. These results provide no advantage over the fixed policies.

Comparator implementation and tuning. COSA and TAFAS use their released update rules and the common observation cutoff. Each receives 12 validation candidates: COSA crosses feedback mode, learning rate $\{10^{-4}, 10^{-3}, 10^{-2}\}$, and update steps $\{1, 3\}$; TAFAS crosses feedback mode, the same learning rates, and weight decay $\{0, 10^{-4}\}$. Every candidate starts from seed 19 and the shared source checkpoint. Selection uses validation MSE, then MAE, and the selected setting remains fixed on test.

M A CHRONOLOGICAL FORECASTING MECHANISM STUDY

This simulation complements the residual-regression experiment in Appendix I. It generates one chronological outcome series, issues forecasts from available sensor readings, and stores residuals after complete delayed feedback. The experiment measures how outcome alignment behaves when forecasts share a realized time series and differ in their available information.

Protocol. The simulator combines an AR(1) sensor, a sinusoidal outcome, and a misspecified frozen forecaster. Its prespecified 2^5 grid crosses memory stride $\{1, 6\}$, observation noise $\{.1, 1\}$, sensor refresh $\{1, 12\}$, coefficient drift off/on, and extra outcome delay $\{0, 24\}$. Eight paired seeds give 256 streams, each scored at the same 128 origins and 24 horizons. Episodes enter FIFO32 only after all targets arrive. Frozen, mean, basis ridge, OARM, count weighting, and timestamp-shuffled groups share the bank, refit schedule, and fixed $\lambda = .1, \beta = .5$. The Full Experimental Record, Section P, gives the generator equations and all condition-level results.

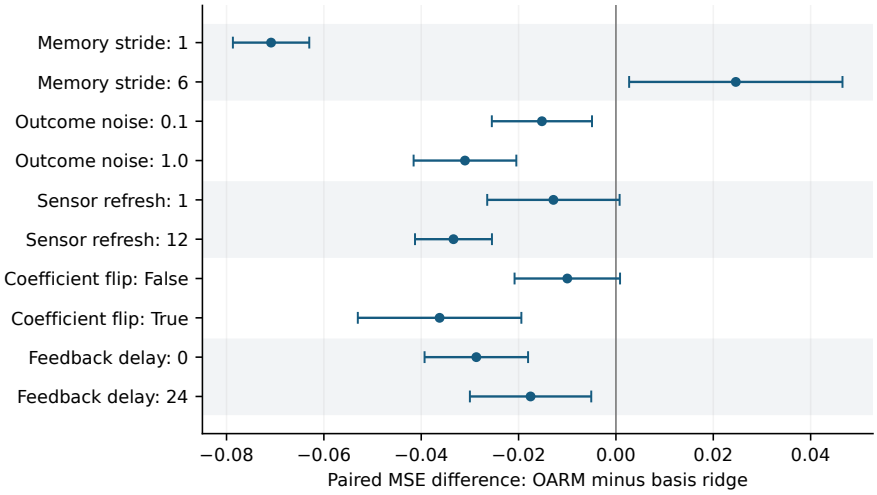

Figure 5: Factorial marginal contrasts in the chronological simulator. Negative favors OARM over basis ridge. Other factors are averaged within each seed before computing an approximate $1.96\,\mathrm{SE}$ interval across eight seeds. Stride jointly changes overlap, memory age, and refit frequency.

Results and verification. OARM has lower MSE than basis ridge in 19/32 conditions: 16/16 at stride one and 3/16 at stride six. The equal-condition difference is -0.023109, with descriptive interval $[-0.033475, -0.012743]$; dense- and sparse-memory contrasts are -0.070846 and $+0.024629$. Extra feedback delay reduces the average advantage from -0.028681 to -0.017536. Macro MSE is 1.496095 for OARM, 1.519204 for basis ridge, and 1.501551 for count weighting. Tests verify shared outcomes, delayed admission, group precision, common evaluation counts, and fitted-system residual below 10^{-8}.