SCOPE: Structure- and Importance-aware Projection for Time-Series Imputation

Anonymous authors
Paper under double-blind review

Abstract

Contiguous gaps disrupt the historical inputs of time-series foundation models (TSFMs), while accurate pointwise reconstruction alone can leave forecasting-relevant structure distorted. We present SCOPE, a model-agnostic imputation method that separates numerical recovery from structural refinement without training a dedicated imputer. Component-wise initialization recovers trend, seasonality, and residuals and supplies the main forecasting gains. Six observation-derived structural targets and positional importance then guide a small, bounded projection, with calibration and fallback to protect the initialization. Across 27 public datasets, four TSFMs, and nine missingness settings, SCOPE achieves the best combined average rank against deep baselines under two training budgets and against classical references. It reduces downstream MAE by 6.97% relative to linear interpolation and improves on native or default handling for three of four TSFMs. Relative to initialization, projection reduces mean structural error by 5.49%. A paired block-length-50 study with fixed imputations supports average forecast non-inferiority within a prespecified 0.5% MAE margin. These results position projection as fidelity-oriented refinement that preserves average predictive performance in the evaluated setting.

Keywords: Time-series foundation models; imputation; structural preservation; importance-weighted projection.

1 Introduction

Pretrained time-series foundation models enable zero-shot forecasting across datasets (Das et al., 2024; Liu et al., 2025; Woo et al., 2024; Ansari et al., 2024; Goswami et al., 2024), but handling incomplete histories remains model-dependent: pipelines may use native missing-value support, default preprocessing, or an external imputer. Contiguous gaps disrupt local continuity, seasonal phase, and trend continuation. A model may extrapolate imputation artifacts as if they were genuine patterns. Upstream imputation should therefore produce histories that support downstream forecasting; MAE or MSE at missing points alone measures only pointwise reconstruction accuracy.

Figure 1 illustrates this objective mismatch. For the same window and TSFM, four methods have similar imputation errors, while forecast MSE ranges from 7.60 to 338, a 44.5-fold difference. Across 6,480 rolling windows, the method with the lowest reconstruction error also achieves the lowest forecast error in only 18.3% of windows. Similar reconstruction errors can conceal different structural deviations. Moreover, equal-magnitude perturbations induce forecast-error increments that differ by approximately 8.7-fold between the two ends of a missing block. Absolute distance to the forecast origin is the main factor, whereas common imputers allocate nearly uniform adjustments within the gap.

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.

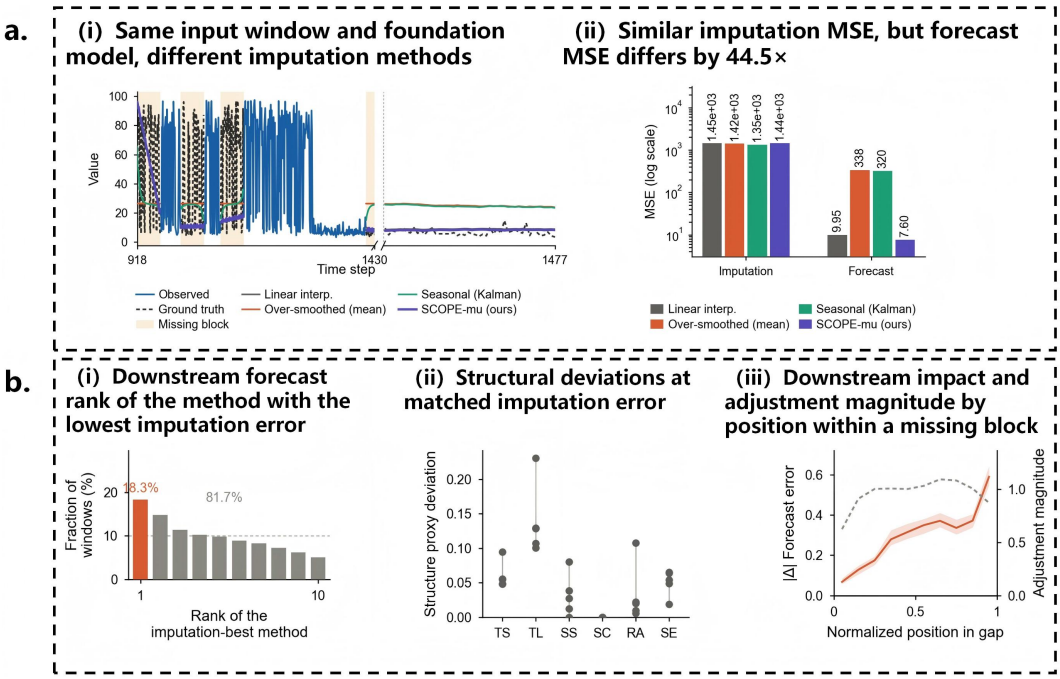

Figure 1: Imputation accuracy and downstream forecasting are not equivalent. Top: imputation and forecasting for one window. Bottom: downstream ranks across windows, structural deviations at matched imputation error, and the downstream effects and adjustment magnitudes of missing positions.

SCOPE separates two responsibilities: M3 performs numerical recovery, while M1, M2, and M4 refine structural fidelity with bounded changes around that completion. This motivates three contributions: (1) six structural targets estimated from observations, separating posterior-averaged shape from observed texture; (2) an importance-weighted projection that balances downstream-use importance against structural leverage, with calibration, observation consistency, and fallback; and (3) evidence separating reconstruction strength from refinement quality, including native-handling comparisons, paired structural evaluation, forecast non-inferiority, and matched-seed analysis. The default method requires no target-TSFM feedback; model-aware alignment is optional.

2 Related Work

Statistical rules, state-space models, and seasonal decomposition offer inexpensive reconstruction but can smooth away phase and local variation in long gaps (Kalman, 1960; Rauch et al., 1965; Moritz and Bartz-Beielstein, 2017; Cleveland et al., 1990; Bandara et al., 2025). Learned imputers include BRITS, SAITS, and CSDI (Cao et al., 2018; Du et al., 2023; Tashiro et al., 2021); ImputeFormer incorporates low-rank structure (Nie et al., 2024), while TimeMixer and TimesNet model multiscale or periodic variation (Wang et al., 2024; Wu et al., 2023). Structural modeling is thus already valuable for recovery. SCOPE adds a separate, selectively activated correction around a strong initialization, jointly using structural targets, positional costs, and bounded updates. Native missing-value handling offers another model-specific alternative, evaluated directly in Section 4.3. Appendix A expands the comparison.

3 Problem Formulation and Method

3.1 Task, Notation, and Overview

Let $X_{\text{obs}} \in \mathbb{R}^{T\times D}$ be an incomplete series with observation mask M, where $M_{t,d} = 1$ denotes an observed entry, and let $\mathcal{G}$ denote the missing positions. We construct a complete input $\widehat{X}$ that

preserves observations for forecasting with a frozen TSFM. With the M3 initialization $\widetilde{X}$, we write $\widehat{X} = \widetilde{X} + \delta$, where $\delta_{t,d} = 0$ at observed positions. The terminal test segment is removed before imputation; T denotes the retained history length. Evaluation labels never enter preprocessing.

Figure 2 separates numerical recovery and structural refinement. M3 first reconstructs a complete history $\widetilde{X}$. M1 estimates a structural target ϕ_{target}, M2 estimates downstream-use importance ω^{use}, structural leverage ω^{lev}, and positional cost r, and M4 applies a bounded correction around $\widetilde{X}$. These three projection modules seek better structural fidelity while preserving the predictive utility already supplied by M3. The default SCOPE-mu returns M3 when pseudo-missing calibration does not support projection; SCOPE-ma optionally adds M5. Algorithm 1 gives the execution order, and Appendix B gives implementation details.

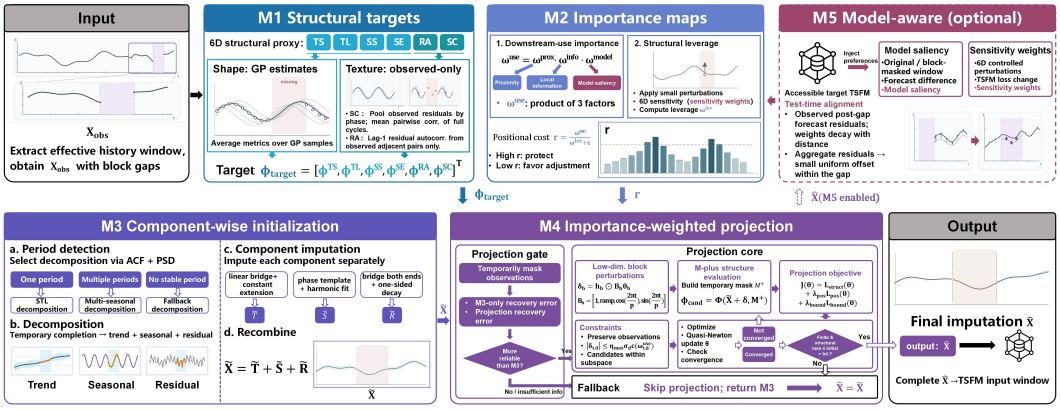

Figure 2: The SCOPE framework. M3 supplies numerical recovery; M1, M2, and M4 guide selective structural refinement with bounded corrections and fallback. M5 optionally adds model-aware alignment.

3.2 M1: Estimating Structural Targets from Observations

The proxy Φ maps a window to six dimensions:

$$\phi = [\text{TS}, \text{TL}, \text{SS}, \text{SE}, \text{RA}, \text{SC}]^\top, \tag{1}$$

representing trend strength, trend linearity, seasonal strength, spectral entropy, lag-one residual autocorrelation, and seasonal correlation. TS and SS use variance ratios of decomposed components; TL measures the linear contribution in an orthogonal linear–quadratic regression of the trend; SC measures consistency across complete periods; RA measures adjacent-residual dependence; and SE is the Shannon entropy of the normalized power spectrum. Definitions appear in Appendix B.2.

A single completion affects the proxy families differently. Near gaps, unstable decomposition can underestimate TS and SS, linear bridging can inflate TL, and boundary discontinuities can shift SE through spectral leakage. Synthetic bridges can also create spurious periodic consistency and adjacent-residual correlation. SCOPE therefore separates posterior averaging of shape descriptors from observation-based estimation of texture.

Computing structure from a single smooth completion embeds its assumptions into the target. For TS, TL, SS, and SE, SCOPE instead samples from a GP posterior fitted only to observed points, using a squared-exponential kernel, a periodic kernel when identifiable, and a noise kernel. Missing-value samples are combined with actual observations into candidate windows $x^{(j)}$, giving

$$\phi_{\text{GP}} = \frac{1}{N}\sum_{j=1}^{N} \Phi(x^{(j)}) \approx \mathbb{E}_{\text{GP}}[\Phi(X) \mid X_{\mathcal{O}}]. \tag{2}$$

We average structural measurements, rather than measure the posterior-mean series. Under the fitted GP and finite second moments, this conditional expectation is optimal for squared-error estimation; performance on real series still depends on model fit.

SC and RA use only observed information. Let S_i denote complete periods of the phase-aggregated seasonal template, r_t observed residuals with mean $\bar{r}$, $\mathcal{O}$ the observed positions, and $\mathcal{P} = \{t : m_t = m_{t-1} = 1\}$ valid adjacent pairs. Then

$$\phi^{\text{SC}} = \text{mean}_{i<j}\,\text{Corr}(S_i, S_j), \qquad \phi^{\text{RA}} = \frac{\sum_{t\in\mathcal{P}}(r_t - \bar{r})(r_{t-1} - \bar{r})}{\sum_{t\in\mathcal{O}}(r_t - \bar{r})^2}. \tag{3}$$

Unidentifiable periods or insufficient valid periods or pairs yield a conservative default of zero. Missing segments do not enter these statistics, so artificial bridges cannot inflate their correlations. Combining these estimates with the four GP estimates gives ϕ_{target}; GP samples estimate the structural target, while M3 performs pointwise recovery.

M1 and M4 use the effective window ending at the retained-history boundary:

$$L = \min(T, nH + C_{\max}), \qquad W_{\text{act}} = X[T - L : T], \tag{4}$$

Here T is the test-start index after truncation, H the horizon, n the rolling-window count, and $C_{\max}$ the maximum context. Split size and the rolling limit determine n as index metadata, without accessing test values. Thus $W_{\text{act}} \subseteq [0, T)$; neither targets nor structural loss use the test suffix. The window prevents distant history from diluting nearby gaps.

3.3 M2: Separating Downstream Risk from Adjustment Leverage

Downstream-use importance measures the potential forecasting risk of a positional error; structural leverage measures how strongly a small change can alter structure:

$$\omega^{\text{use}}_{t,d} = \omega^{\text{prox}}_{t,d}\,\omega^{\text{info}}_{t,d}\,\omega^{\text{model}}_{t,d}, \quad \omega^{\text{lev}}_{t,d} = \sum_{k=1}^{6} v_k \left|\frac{\partial \Phi_k}{\partial x_{t,d}}\right|, \quad r_{t,d} = \frac{\omega^{\text{use}}_{t,d}}{\omega^{\text{lev}}_{t,d} + \varepsilon}. \tag{5}$$

The proximity factor decays exponentially with distance from the forecast origin. Local information combines local variance, first-difference magnitude, and a change-point posterior (Adams and MacKay, 2007). The model factor defaults to one, and structural weights v_k are uniform and sum to one. Finite differences are evaluated at sampled missing positions and interpolated elsewhere. During evaluation, the perturbed position is temporarily marked as observed so that internal interpolation does not overwrite the perturbation. Each map is robustly quantile-clipped and mapped to the unit interval. A large r indicates high downstream risk with limited structural effect and discourages modification; a small r favors adjustment.

Locally, proxy changes can be approximated linearly around the current completion, so higher leverage indicates a stronger structural response to a given adjustment. Use importance alone can overprotect downstream-relevant positions, whereas leverage alone can favor forecast-sensitive positions. Their ratio balances both concerns: high-r positions incur a greater penalty for large corrections, while low-r positions can carry more of the repair. This is a local weighting principle for the projection objective. The finite-difference evaluation uses the same temporary mask for baseline and perturbed values, and a lower bound on normalized leverage avoids numerical singularities.

3.4 M3: Component-wise Initialization

M3 is the primary numerical recovery stage and constructs the forecasting input around which subsequent projection operates. The autocorrelation function (ACF) and power spectral density (PSD) of a temporary completion jointly guide period detection: STL (Cleveland et al., 1990) handles a single period, MSTL (Bandara et al., 2025) handles multiple periods, and unstable periodicity triggers a trend–residual decomposition. Temporary component values inside gaps are re-estimated according to the mask rather than treated as observations.

For an interior gap $[s, e)$, trend and residual components independently connect their observed boundaries:

$$\widetilde{C}(t) = C(s-1) + \frac{t - (s-1)}{e - (s-1)}[C(e) - C(s-1)], \quad C \in \{T, R\}, \quad t \in [s, e). \tag{6}$$

At endpoints, the trend extends the nearest boundary value, whereas residuals decay toward zero. Seasonality combines a phase template fitted to observations and a low-order harmonic regression:

$$\widetilde{S} = \alpha_p S_{\text{pat}} + (1 - \alpha_p) S_{\text{harm}}. \tag{7}$$

The weight α_p increases with observed phase coverage, and the period length limits harmonic order. The seasonal component is zero when the period is unidentifiable or fewer than one period of observations are available. We form $\widetilde{X} = \widetilde{T} + \widetilde{S} + \widetilde{R}$ and restore observations. Sparse or entirely missing columns receive conservative handling without forced decomposition. The components separately recover low-frequency shape, periodic phase, and boundary continuity; Appendix B.3 gives diagnostic confidence bands and further fallback details.

3.5 M4: Importance-weighted Structural Projection

Low-dimensional candidates and objective. M4 refines the M3 completion toward M1 targets; M2 penalizes changes at important positions. Its role is structural refinement with limited disturbance to numerical recovery. For each gap b, a Hann envelope h_b and low-frequency basis generate candidate perturbations. All blocks are optimized jointly within the same effective window:

$$\delta_b = h_b \odot (B_b \theta_b), \qquad B_b = [\mathbf{1}, \text{ramp}, \cos(2\pi t/p), \sin(2\pi t/p)]. \tag{8}$$

A typical block has four parameters controlling offset, slope, and first-order periodic variation. The objective is

$$J(\theta) = L_{\text{struct}} + \lambda_{\text{pos}} L_{\text{pos}} + \lambda_{\text{bound}} L_{\text{bound}}, \tag{9}$$

$$L_{\text{struct}} = \sum_{k=1}^{6} w_k \left(\frac{\Phi_k(\widetilde{X} + \delta, M^+) - \phi_k^*}{|\phi_k^*| + c} \right)^2, \quad L_{\text{pos}} = \sum_{(t,d)\in\mathcal{G}} r_{t,d} \delta_{t,d}^2, \tag{10}$$

where ϕ_k^* is a target component, w_k defaults to uniform weights, and L_{bound} softly penalizes amplitude violations. Positional costs favor lower-risk positions with greater structural leverage.

M-plus evaluation and constraints. Using the original mask inside the structural function would overwrite candidate perturbations through internal interpolation and remove useful optimization directions. M4 copies the mask and temporarily marks the optimized gaps as observed, obtaining M^+ so that Φ reads candidate values. The original mask and M1 targets remain unchanged. Perturbations affect only missing positions in the effective window and satisfy

$$|\delta_{t,d}| \le \eta_{\max} \sigma_d c(\omega_{t,d}^{\text{use}}), \tag{11}$$

where $c(\cdot)$ decreases with use importance. L-BFGS-B optimization (Byrd et al., 1995) starts at $\theta = 0$, with finite-difference gradients in parameter space. After optimization, perturbations are clipped pointwise, observations are restored, and structural loss is checked. The low-dimensional parameterization constrains candidates before clipping; the final correction need not remain in that subspace.

Calibration, acceptance, and prediction preservation. The projection gate compares M3 and projection on two pseudo-missing blocks per input, sampled from observed history before T. Full projection is activated only when the calibration recovery gain strictly exceeds 0.25%; insufficient information returns M3. This gate uses recovery evidence without accessing future forecast labels. After optimization and clipping, a candidate is accepted only if finite and its estimated-target structural loss is no higher than M3's within tolerance, using the same M^+, period, and weights. Otherwise M3 is returned. These safeguards control the correction; prediction preservation is assessed separately by paired non-inferiority tests. Here, projection denotes constrained structural correction rather than orthogonal projection or exact target matching. Appendix C.5 gives numerical settings.

3.6 M5: Optional Model-aware Alignment with Fallback

When the target TSFM is accessible, M5 applies controlled structural perturbations on historical calibration windows preceding T. Forecast-loss increments, clipped below at zero and temperature-normalized, yield model-sensitivity weights that replace the uniform weights in M2 and M4.

Let Δ_k be the calibration-average difference between forecast loss under a controlled perturbation of structural dimension k and loss on the original history. For temperature τ, the calibrated weights are

$$w_k^{\text{model}} = \frac{\exp(\max(\Delta_k, 0)/\tau)}{\sum_{\ell=1}^{6} \exp(\max(\Delta_\ell, 0)/\tau)}. \tag{12}$$

These nonnegative weights sum to one and direct both structural correction and positional costs toward model-sensitive dimensions.

A block-occlusion saliency interface is also available but was not enabled in the M5 experiments. After projection, actual visible observations following a gap but preceding T can provide distance-weighted, clipped forecast residuals for a small uniform additive offset within the block. Insufficient information leaves the result unchanged. M5 is enabled only when calibration supports a net benefit; otherwise it falls back to SCOPE-mu. Appendix B.5 details the formulas and information used.

3.7 Execution and Computational Cost

Algorithm 1 makes the gating and fallback order explicit. Z denotes standardized history; μ and σ are estimated from observed values. Subroutines implement Sections 3.2–3.6. Acceptance compares both losses with the same W, M^+, period, and weights; ε_{acc} is the acceptance tolerance.

Algorithm 1 SCOPE: numerical recovery and gated structural projection

Input: Retained history $X_{\text{obs}}[0:T]$, mask M, proxy Φ, task metadata $(H, n, C_{\max})$
Input: Optional TSFM g and calibration set $\mathcal{C}$ entirely before T
Output: Complete history $\widehat{X}$ with $\widehat{X}[M] = X_{\text{obs}}[M]$

$(Z, \mu, \sigma) \leftarrow \textsc{StandardizeObserved}(X_{\text{obs}}, M)$
$(\widetilde{Z}, p) \leftarrow \text{M3}(Z, M)$; $\widetilde{Z}[M] \leftarrow Z[M]$; $\widehat{Z} \leftarrow \widetilde{Z}$
$a \leftarrow \textsc{CalibrationGain}(Z, M, \Phi, 2)$ ▷ Two pseudo-missing blocks
if a is available **and** $a > 0.25\%$ **then**
  $L \leftarrow \min(T, nH + C_{\max})$; $W \leftarrow [T - L, T)$
  $\phi^* \leftarrow \text{M1}(Z[W], M[W], \Phi, p)$; $w \leftarrow \mathbf{1}/6$
  if M5 is enabled and supported by historical calibration **then**
    $w \leftarrow \textsc{ModelWeights}(g, \mathcal{C})$ ▷ Equation 12
  end if
  $(\omega^{\text{use}}, \omega^{\text{lev}}, r) \leftarrow \text{M2}(\widetilde{Z}, M, W, \Phi, w, p)$
  $(h, B, M^+) \leftarrow \textsc{ProjectionSetup}(M, W, p)$
  $\theta^* \leftarrow \text{LBFGSB}(J(\theta; \widetilde{Z}, \phi^*, r, w, M^+), \theta_0 = 0)$
  $\delta \leftarrow \textsc{ClipToBounds}(h \odot (B\theta^*), \omega^{\text{use}})$; $\delta[M] \leftarrow 0$
  $Z_{\text{cand}} \leftarrow \widetilde{Z} + \delta$
  if Z_{cand} is finite **and** $L_{\text{struct}}(Z_{\text{cand}}) \le L_{\text{struct}}(\widetilde{Z}) + \varepsilon_{\text{acc}}$ **then**
    $\widehat{Z} \leftarrow Z_{\text{cand}}$ ▷ Otherwise retain M3
  end if
  if M5 is enabled and historical calibration supports alignment **then**
    $\widehat{Z} \leftarrow \textsc{Align}(\widehat{Z}, Z, M, g, \mathcal{C})$ ▷ Visible values before T only
  end if
end if
$\widehat{X} \leftarrow \mu + \sigma \odot \widehat{Z}$; $\widehat{X}[M] \leftarrow X_{\text{obs}}[M]$
return $\widehat{X}$

The principal costs are GP fitting and structural evaluation in M1, sensitivity estimation in M2, component recovery in M3, and low-dimensional iterative optimization in M4. Optimization dimension grows with the number of gaps, while proxy evaluation depends on window length. M3 is typically less costly than GP fitting or TSFM inference. Selective activation avoids full projection when calibration does not support it, avoiding the corresponding full optimization calls. Optional M5 adds model calibration and queries; the default pipeline requires no dedicated imputation-model training.

4 Experiments

4.1 Setup and Evaluation

We evaluate 27 public datasets spanning energy, the environment, transportation, economics and finance, healthcare, and cloud operations. Frozen backbones are Chronos-2 (Ansari et al., 2025), Kairos-23M (Feng et al., 2026), Sundial (Liu et al., 2025), and TimesFM-2.5 (Google Research, 2025); the optional M5 study adds VisionTS++ (Shen et al., 2025). Variables are evaluated as uni-

variate series. Missing rates $\{10\%, 20\%, 30\%\}$ and block lengths $\{50, 100, 150\}$ give 243 inputs and 972 model configurations. Each configuration uses 20 rolling windows, with stride H and horizons of 48, 30, and 8 for minute/hour, daily, and weekly data, respectively.

We first remove the terminal test segment $X[T : T_{\text{full}}]$, then impute only the retained incomplete history $X_{\text{obs}}[0 : T]$. All rolling forecast origins satisfy $o_j \geq T$ (Aksu et al., 2024). Normalization, M1–M4, gating calibration, and M5 access only retained historical observations; each TSFM call uses history strictly before o_j. Test targets are reserved for scoring. All methods share masks, origins, and TSFMs. Appendix C.3 specifies the index and data-access boundaries.

The primary metric is downstream MAE, supplemented by MSE and sMAPE. MAE and sMAPE use predictive medians; MSE evaluates the point forecast returned by each model interface, which need not be a predictive mean. For aggregation, each error is divided by the same reference method's error within each model–dataset–missingness configuration, and relative errors are averaged with equal weights. This prevents large-scale datasets from dominating. We also report average ranks and configuration-wise wins and losses. Baseline-5 contains five deep methods evaluated under equal and full budgets, the latter increasing training epochs. Reference-9 contains SCOPE-mu and eight classical methods. The larger budget tests whether the ordering persists with additional baseline training. New confidence intervals use 10,000 dataset-cluster bootstrap draws, preserving within-dataset dependence. Fixed-imputation paired ablation and forecast non-inferiority use block length 50 (81 inputs; 324 model configurations).

4.2 Forecasting Performance and Execution Cost

Table 1: Average ranks under two deep-baseline training budgets (lower is better). Each budget compares the same six methods.

	Equal budget				Full budget			
Method	MAE	MSE	sMAPE	Avg.	MAE	MSE	sMAPE	Avg.
SCOPE-mu	**1.59**	**1.82**	**1.78**	**1.73**	**1.69**	**1.96**	**1.86**	**1.83**
ImputeFormer	2.73	2.96	2.75	2.81	2.72	2.75	2.79	2.75
CSDI	3.21	3.14	3.25	3.20	3.01	2.97	2.91	2.96
TimesNet	4.32	4.16	4.24	4.24	4.56	4.45	4.44	4.48
TimeMixer	4.53	4.41	4.52	4.49	4.77	4.57	4.69	4.68
BRITS	4.62	4.51	4.47	4.53	4.25	4.30	4.32	4.29

Table 1 shows that SCOPE-mu ranks first under both deep-baseline budgets, with combined average ranks of 1.73 and 1.83. Its configuration-wise MAE win rates against the five equal-budget deep baselines range from 80.8% to 95.1%. Additional training improves some baselines, but SCOPE-mu retains the best combined rank. In Reference-9, its combined rank is 2.46, followed by Linear at 3.56 and GP-RBF at 4.16. Against Linear, it records 548 wins, 180 losses, and 244 ties: a 75.3% win rate excluding ties, wins in 56.4% of all configurations, and a 6.97% mean relative MAE reduction. Against STL-Kalman, it records 686 wins, 258 losses, and 28 ties, with a 72.7% win rate excluding ties and a 10.43% mean relative MAE reduction. These are aggregate advantages; reference methods still achieve lower errors on some datasets. Full results appear in Appendix D.

Timing uses first-execution records without cache hits for 81 configurations: Kairos-23M, 27 datasets, three missing rates, and block length 50. It includes imputation, per-input training of deep methods, Kairos forecasting, metric computation, and I/O, but **excludes TSFM preloading and precomputed SCOPE gating calibration**. SCOPE-mu takes 4.02 seconds per configuration. Under equal budgets, TimeMixer and ImputeFormer take 2.07 and 3.12 seconds, while the other deep methods take 4.38–37.14 seconds. Under full budgets, deep methods take 5.00–88.11 seconds. These measurements characterize the recorded execution stage.

Figure 3 relates execution cost to MAE win rates: 80.8%, 81.1%, 92.8%, 91.5%, and 95.1% against CSDI, ImputeFormer, TimeMixer, TimesNet, and BRITS. Additional training reduces normalized MAE from 123.2 to 113.0 for ImputeFormer, 133.7 to 129.6 for CSDI, and 161.6 to 154.9 for BRITS, narrowing some gaps at increased cost without changing SCOPE-mu's leading combined rank. Thus

the ranking is robust to the tested increase in training budget; comparisons remain specific to these implementations and budgets.

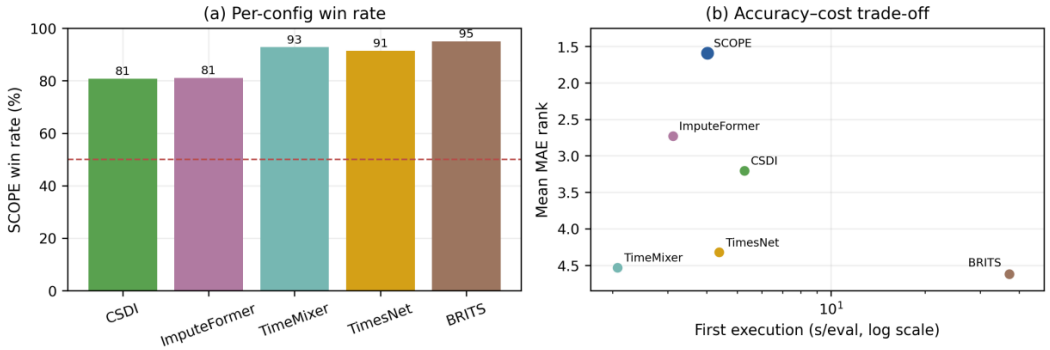

Figure 3: Equal-budget deep baselines: configuration-wise MAE win rates and the accuracy–execution-cost tradeoff. Timing uses the 81-configuration Kairos subset and excludes TSFM preloading and precomputed gating calibration.

4.3 Comparison with Native or Default Missing Handling

Table 2: Native/default comparison: 243 configurations per model, with baseline MAE normalized to 100. Positive gain favors SCOPE.

Model	Comparator	SCOPE MAE	Gain (%)	95% CI of gain
Chronos-2	Native	104.95	−4.95	[−13.93, 2.24]
Kairos-23M	Native	86.71	13.29	[7.70, 19.70]
TimesFM-2.5	Native	84.63	15.37	[10.89, 20.13]
Sundial	Default	86.36	13.64	[9.55, 17.87]

SCOPE improves normalized MAE over TimesFM-2.5, Kairos, and Sundial by 15.37%, 13.29%, and 13.64%, respectively (Table 2); all three dataset-bootstrap intervals exclude zero. Chronos-2 favors native handling: SCOPE's mean MAE is 4.95% higher, while its median relative deficit is 0.41%. This identifies a model-specific strength of Chronos-2. SCOPE offers a shared preprocessing method across frozen models, without learning a separate imputer or using model feedback. Its cross-model utility is supported by the other three backbones, while native Chronos-2 remains preferable on average in this comparison. Appendix D.4 reports seeds, medians, intervals, and wins/losses.

4.4 Numerical Recovery and Structural Refinement

Figure 4 contrasts recovery and projection. Removing M3 raises mean MAE, MSE, and sMAPE by 7.24%, 16.16%, and 4.99%, respectively (a). Removing M1, M2, or M4 raises MAE by 0.37%, 0.11%, or 0.08%; MSE and sMAPE also increase (Appendix E.4). M3 supplies the main numerical recovery; the projection stage is evaluated collectively through structural fidelity and average forecast preservation.

Table 3: Structural fidelity before and after projection. Errors average absolute deviations across six dimensions; dataset-bootstrap intervals describe the absolute decrease.

Inputs	M3 error	SCOPE error	Decrease	95% CI of decrease
All 243	0.045734	0.043222	0.002512	[0.000786, 0.004977]
Active 97	0.052680	0.046387	0.006293	[0.002150, 0.012619]

Projection reduces mean six-dimensional structural error by 5.49% across all 243 inputs and 11.95% among the 97 activated inputs (Figure 4b). Both absolute decreases have strictly positive 95% dataset-bootstrap intervals (Table 3). Dataset-level wins/losses/ties are 20/4/3; sign and Wilcoxon tests give $p = 0.0015$ and $p = 0.0005$. Gains mainly involve TL, TS, SS, and SE; RA is slightly worse. Clean features are used only for post-hoc evaluation, separately from M4's estimated-target acceptance loss.

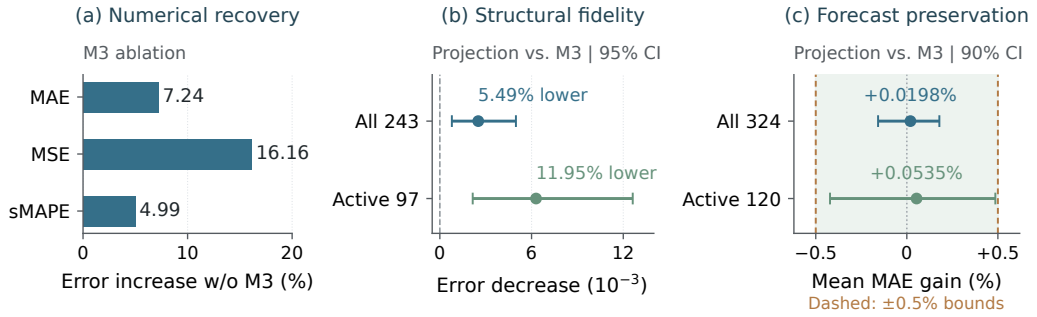

Figure 4: Recovery, fidelity, and prediction preservation. (a) Forecast error increases without M3 in the original block-length-50 ablation. (b) Structural error decreases over 243 inputs with 95% intervals; annotations show relative decreases. (c) Mean MAE gains over 324 configurations with 90% intervals at block length 50, with fixed imputations and gates; dashed lines mark $\pm 0.5\%$ bounds. Intervals in (b,c) use dataset-cluster bootstrap; positive values favor projection.

In the matched-seed block-length-50 comparison, projection's mean MAE gain over M3 is 0.0198%, with a 90% interval of $[-0.1578\%, 0.1787\%]$. Both full-sample and active-group intervals lie within $\pm 0.5\%$ (Figure 4c), satisfying the prespecified average non-inferiority and equivalence criteria. These conclusions condition on fixed imputations and gates; Appendix E.6, Table 25, gives full results and imputation-seed variation.

Selective activation and inference noise. The gate activates 97/243 inputs (39.92%): 78 improve structurally and 19 worsen; all 146 inactive inputs remain unchanged. Projection's mean adjustment is 2.96% of Linear→M3 initialization over actual forecast contexts. Under five matched Sundial seeds, M1, M2, and M4 effects are 9.92, 6.22, and 6.41 times the paired inference standard deviation, while four-model dataset-level tests remain nonsignificant after Holm correction. The evidence supports collective structural refinement and average forecast preservation, with per-input variation (Appendices E.7–E.8).

Scenarios and optional model alignment. SCOPE has the lowest mean relative MAE among seven methods in all nine missingness groups, with a 13.4% gain over Linear under strong seasonality and little difference under weak seasonality (Appendix E). Optional M5 addresses model-specific preferences: on VisionTS++, it lowers mean MAE by 4.79% relative to SCOPE-mu, with a 75.3% win rate. On the four main backbones, fallback retains SCOPE-mu in all configurations (Appendix F).

5 Conclusion and Limitations

SCOPE separates component-wise numerical recovery from selective structural refinement. M3 supplies the main forecasting gains; M1, M2, and M4 improve average structural fidelity with small bounded corrections and empirical average prediction preservation. Results support broad utility across frozen TSFMs, while Chronos-2's native handling remains stronger on average. Limitations include proxy estimation, weak-seasonality cases, and model-dependent M5 calibration. Forecast non-inferiority is established for the tested fixed-imputation, block-length-50 comparison; robustness across imputation seeds and longer gaps remains open.

AI Use Statement

AI tools assisted with language editing, translation, and document formatting.

Reproducibility Statement

The code and original data for this work are submitted in the Supplementary Material. Appendix C.5 lists the experimental settings. Section 3 specifies the method, constraints, and acceptance criteria. Appendix B provides the algorithm and implementation conventions; Appendix C specifies data, models, and evaluation; and Appendices D–F report full results and supplementary analyses.

References

Adams, R. P., & MacKay, D. J. C. (2007). Bayesian online changepoint detection. arXiv preprint arXiv:0710.3742. https://arxiv.org/abs/0710.3742.

Aksu, T., Woo, G., Liu, J., Liu, X., Liu, C., Savarese, S., Xiong, C., & Sahoo, D. (2024). GIFT-Eval: A benchmark for general time series forecasting model evaluation. arXiv preprint arXiv:2410.10393. https://arxiv.org/abs/2410.10393.

Ansari, A. F., Stella, L., Türkmen, A. C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Pineda-Arango, S., Kapoor, S., Zschiegner, J., Maddix, D. C., Wang, H., Mahoney, M. W., Torkkola, K., Wilson, A. G., Bohlke-Schneider, M., & Wang, B. (2024). Chronos: Learning the Language of Time Series. Transactions on Machine Learning Research. https://openreview.net/forum?id=gerNCVqqtR.

Ansari, A. F., Shchur, O., Küken, J., Auer, A., Han, B., Mercado, P., Rangapuram, S. S., Shen, H., Stella, L., Zhang, X., Goswami, M., Kapoor, S., Maddix, D. C., Guerron, P., Hu, T., Yin, J., Erickson, N., Desai, P. M., Wang, H., Rangwala, H., Karypis, G., Wang, Y., & Bohlke-Schneider, M. (2025). Chronos-2: From Univariate to Universal Forecasting. arXiv preprint arXiv:2510.15821. https://arxiv.org/abs/2510.15821.

Bandara, K., Hyndman, R. J., & Bergmeir, C. (2025). MSTL: A Seasonal-Trend Decomposition Algorithm for Time Series with Multiple Seasonal Patterns. International Journal of Operational Research, 52(1), 79–98. https://doi.org/10.1504/IJOR.2025.143957.

Byrd, R. H., Lu, P., Nocedal, J., & Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing, 16(5), 1190–1208. https://doi.org/10.1137/0916069.

Candès, E. J., & Recht, B. (2009). Exact matrix completion via convex optimization. Foundations of Computational Mathematics, 9(6), 717–772. https://doi.org/10.1007/s10208-009-9045-5.

Cao, W., Wang, D., Li, J., Zhou, H., Li, L., & Li, Y. (2018). BRITS: Bidirectional recurrent imputation for time series. Advances in Neural Information Processing Systems, 31. https://dblp.org/rec/conf/nips/CaoWLZLL18.

Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent neural networks for multivariate time series with missing values. Scientific Reports, 8, 6085. https://www.nature.com/articles/s41598-018-24271-9.

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on loess. Journal of Official Statistics, 6(1), 3–73. https://www.scb.se/contentassets/ca21efb41fee47d293bbee5bf7be7fb3/stl-a-seasonal-trend-decomposition-procedure-based-on-loess.pdf.

Das, A., Kong, W., Sen, R., & Zhou, Y. (2024). A decoder-only foundation model for time-series forecasting. Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 10148–10167. https://proceedings.mlr.press/v235/das24c.html.

Du, W., Côté, D., & Liu, Y. (2023). SAITS: Self-attention-based imputation for time series. Expert Systems with Applications, 219, 119619. https://doi.org/10.1016/j.eswa.2023.119619.

Feng, K., Lan, S., Fang, Y., He, W., Lu, S., Gu, S., Ma, L., Lu, X., & Ren, K. (2026). Kairos: Toward Adaptive and Parameter-Efficient Time Series Foundation Models. arXiv preprint arXiv:2509.25826v3, May 14, 2026. https://arxiv.org/abs/2509.25826v3.

Fortuin, V., Baranchuk, D., Rätsch, G., & Mandt, S. (2020). GP-VAE: Deep probabilistic time series imputation. Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics, PMLR 108, 1651–1661. https://dblp.org/pid/218/7489.html.

Google Research (2025). TimesFM 2.5. Software release, September 15, 2025. Official TimesFM repository. https://github.com/google-research/timesfm#update---sept-15-2025.

Goswami, M., Szafer, K., Choudhry, A., Cai, Y., Li, S., & Dubrawski, A. (2024). MOMENT: A Family of Open Time-series Foundation Models. Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 16115–16152. https://proceedings.mlr.press/v235/goswami24a.html.

Hewamalage, H., Ackermann, K., & Bergmeir, C. (2023). Forecast evaluation for data scientists: Common pitfalls and best practices. Data Mining and Knowledge Discovery, 37(2), 788–832. https://doi.org/10.1007/s10618-022-00894-5.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Journal of Basic Engineering, 82(1), 35–45. https://doi.org/10.1115/1.3662552.

Little, R. J. A., & Rubin, D. B. (2019). Statistical analysis with missing data (3rd ed.). Wiley. https://uat.store.wiley.com/en-us/statistical-analysis-with-missing-data-3rd-edition-p-9781118595695.

Liu, Y., Yu, R., Zheng, S., Zhan, E., & Yue, Y. (2019). NAOMI: Non-autoregressive multiresolution sequence imputation. Advances in Neural Information Processing Systems, 32. https://dblp.org/rec/conf/nips/LiuYZZY19.

Liu, Y., Qin, G., Shi, Z., Chen, Z., Yang, C., Huang, X., Wang, J., & Long, M. (2025). Sundial: A Family of Highly Capable Time Series Foundation Models. Proceedings of the 42nd International Conference on Machine Learning, PMLR 267, 39295–39317. https://proceedings.mlr.press/v267/liu25be.html.

Mazumder, R., Hastie, T., & Tibshirani, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. Journal of Machine Learning Research, 11, 2287–2322. https://dblp.org/rec/journals/jmlr/MazumderHT10.

Moritz, S., & Bartz-Beielstein, T. (2017). imputeTS: Time series missing value imputation in R. The R Journal, 9(1), 207–218. https://journal.r-project.org/articles/RJ-2017-009/.

Nie, T., Qin, G., Ma, W., Mei, Y., & Sun, J. (2024). ImputeFormer: Low Rankness-Induced Transformers for Generalizable Spatiotemporal Imputation. Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2260–2271. https://doi.org/10.1145/3637528.3671751.

Qiao, Z., Pan, S., Wang, A., Zhukova, V., Liu, Y., Jiang, X., Wen, Q., Long, M., Jin, M., & Liu, C. (2026). It's TIME: Towards the next generation of time series forecasting benchmarks. International Conference on Machine Learning. Camera-ready version, arXiv:2602.12147v4. https://arxiv.org/abs/2602.12147v4.

Rauch, H. E., Tung, F., & Striebel, C. T. (1965). Maximum likelihood estimates of linear dynamic systems. AIAA Journal, 3(8), 1445–1450. https://doi.org/10.2514/3.3166.

Rubin, D. B. (1976). Inference and missing data. Biometrika, 63(3), 581–592. https://academic.oup.com/biomet/article-abstract/63/3/581/270932.

Shen, L., Chen, M., Liu, X., Fu, H., Ren, X., Sun, J., Li, Z., & Liu, C. (2025). VisionTS++: Cross-Modal Time Series Foundation Model with Continual Pre-trained Vision Backbones. arXiv preprint arXiv:2508.04379v3. https://arxiv.org/abs/2508.04379v3.

Stekhoven, D. J., & Bühlmann, P. (2012). MissForest: Non-parametric missing value imputation for mixed-type data. Bioinformatics, 28(1), 112–118. https://doi.org/10.1093/bioinformatics/btr597.

Tashiro, Y., Song, J., Song, Y., & Ermon, S. (2021). CSDI: Conditional score-based diffusion models for probabilistic time series imputation. Advances in Neural Information Processing Systems, 34, 24804–24816. https://dblp.org/rec/conf/nips/TashiroSSE21.

Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. International Journal of Forecasting, 16(4), 437–450. https://doi.org/10.1016/S0169-2070(00)00065-0.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., & Altman, R. B. (2001). Missing value estimation methods for DNA microarrays. Bioinformatics, 17(6), 520–525. https://doi.org/10.1093/bioinformatics/17.6.520.

van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. Journal of Statistical Software, 45(3), 1–67. https://www.jstatsoft.org/article/view/v045i03.

Wang, S., Wu, H., Shi, X., Hu, T., Luo, H., Ma, L., Zhang, J. Y., & Zhou, J. (2024). TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. International Conference on Learning Representations. https://dblp.org/rec/conf/iclr/WangWSHLMZ024.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., & Sahoo, D. (2024). Unified Training of Universal Time Series Forecasting Transformers. Proceedings of the 41st International Conference on Machine Learning, PMLR 235, 53140–53164. https://proceedings.mlr.press/v235/woo24a.html.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., & Long, M. (2023). TimesNet: Temporal 2D-variation modeling for general time series analysis. International Conference on Learning Representations. https://arxiv.org/abs/2210.02186.

Yoon, J., Zame, W. R., & van der Schaar, M. (2019). Estimating Missing Data in Temporal Data Streams Using Multi-Directional Recurrent Neural Networks. IEEE Transactions on Biomedical Engineering, 66(5), 1477–1490. https://doi.org/10.1109/TBME.2018.2874712.

A Additional Related Work and Method Positioning

Table 4 compares six method families across eight design dimensions. It describes their design emphases, rather than an absolute ranking of empirical performance.

Table 4: Design comparison of temporal imputation methods for TSFM input repair.

Method family	Training and deployment	Primary objective	Block gaps	Struct.	Import.	TSFM alignment	Limitations
Simple rules (mean, forward, linear/spline)	No training; lowest cost	Local continuity and smoothness	●○○	●○○	○○○	○○○	Oversmoothing in long gaps; weak recovery of phase and abrupt changes
State space (Kalman filtering/smoothing, ETS)	No task training; state or parameter estimation	Probabilistic state estimation and smoothing	●●○	●●○	○○○	○○○	Linear, Gaussian, or stationarity assumptions limit complex nonlinear structure
ML regression (MissForest, KNN, XGBoost)	Iterative fitting or feature construction	Nonlinear variable relationships	●○○	●○○	○○○	○○○	Limited temporal order, long-range dependence, and cross-dataset transfer
Deep bidirectional models (BRITS, GRU-D)	Offline training	Reconstruction and bidirectional consistency	●●○	●●○	●○○	●○○	Training-distribution dependence; no explicit TSFM input-structure objective
Transformer/ diffusion models (SAITS, CSDI)	Offline training; costly training and inference	Conditional reconstruction or distribution fitting	●●●	●●○	●○○	●○○	Complex deployment; structure and forecasting suitability usually not separately constrained
SCOPE (ours)	No dedicated model training; optional precomputation and model feedback	Structural preservation and importance-weighted projection	●●●	●●●	●●●	●●●	Depends on structural proxies and positional-importance estimates

●●●: explicit objective or strong support, ●●○: partial, often implicit or assumption-dependent support, ●○○: weak support, ○○○: usually not a design objective. The comparison describes design emphases, not an absolute performance ranking across datasets.

Imputation for TSFMs is closely related to their input mechanisms and evaluation protocols. TimesFM, Sundial, and related models illustrate the shift from dataset-specific training toward cross-domain zero-shot forecasting through large-scale pretraining (Das et al., 2024; Liu et al., 2025; Woo et al., 2024; Ansari et al., 2024; Goswami et al., 2024). GIFT-Eval and TIME additionally emphasize data leakage, task formulation, and structural-pattern analysis beyond average prediction error (Aksu et al., 2024; Qiao et al., 2026; Tashman, 2000; Hewamalage et al., 2023). TIME identifies task formulations disconnected from real-world contexts as a limitation of existing benchmarks (Qiao et al., 2026). Input repair for TSFMs should consequently serve the forecasting task and preserve input structure.

Missing-data inference provides the statistical foundation (Little and Rubin, 2019; Rubin, 1976); mice implements chained-equation imputation for multivariate data (van Buuren and Groothuis-Oudshoorn, 2011). Statistical and state-space imputers offer low deployment costs and interpretability. Mean imputation, forward filling, linear or spline interpolation, and seasonal decomposition quickly restore complete inputs; tools such as imputeTS integrate these approaches with temporal models including Kalman smoothing (Moritz and Bartz-Beielstein, 2017; Kalman, 1960; Rauch et al., 1965). STL provides interpretable trend, seasonal, and residual components that naturally support structural proxies (Cleveland et al., 1990). These methods commonly rely on local smoothness, linear states, or short-term stationarity. Within contiguous gaps, the lack of observed anchors can lead to excessive smoothing of trend changes, seasonal phase, and high-frequency structure.

Machine-learning and low-rank approaches exploit inter-variable relationships or global matrix structure. Similarity-based estimates have been studied for missing microarray values (Troyanskaya et al., 2001). MissForest iteratively estimates missing values using nonparametric random-forest regression, accommodating mixed variable types and nonlinear relationships (Stekhoven and Bühlmann, 2012). SoftImpute formulates recovery as low-rank matrix completion and uses nuclear-norm regularization with soft-thresholded singular-value decomposition for scalable estimation (Mazumder et al., 2010; Candès and Recht, 2009). These methods adapt more flexibly to data than simple rules, but primarily model variable dependence or low-rank structure rather than trends, seasonality, phase, and positions relative to the forecasting window. Long-context TSFM input repair therefore benefits from additional temporal constraints.

Deep temporal imputers incorporate missingness patterns, temporal dependencies, and multivariate correlations into learned representations. GRU-D models informative missingness using masks and time-gap decay (Che et al., 2018). M-RNN combines within-stream interpolation and cross-stream imputation using multidirectional recurrent networks (Yoon et al., 2019). NAOMI instead uses non-autoregressive multiresolution sequence imputation (Liu et al., 2019). BRITS treats missing values as learnable variables in a recurrent graph and uses bidirectional dynamics with consistency constraints (Cao et al., 2018). SAITS employs diagonally masked self-attention to capture temporal dependencies and feature correlations (Du et al., 2023). These expressive models usually require offline training and optimize reconstruction or task losses. Training-distribution dependence and objective mismatch become more consequential under domain shift or when the downstream model is replaced by a black-box TSFM.

Probabilistic generative approaches model uncertainty and produce diverse imputations. GP-VAE assumes a low-dimensional latent representation that evolves smoothly under a GP, combining dimensionality reduction with imputation (Fortuin et al., 2020). CSDI trains conditional diffusion models explicitly for imputation and exploits relationships among observed values (Tashiro et al., 2021). Such distributional estimates are useful under uncertainty, but incur training and sampling costs. Most approaches still prioritize pointwise error or likelihood without explicitly constraining the completed window's suitability for TSFM extrapolation.

B Notation, Algorithm, and Implementation Details

B.1 Notation and Execution Conventions

Table 5 defines the main notation. Algorithm 1 in Section 3.7 specifies the execution flow, including selective activation and fallback.

Table 5: Main notation.

Symbol	Meaning	Definition
$X \in \mathbb{R}^{T\times D}$	Complete ground-truth series	Length T, number of variables D.
$x_{t,d}$	Series entry	Ground-truth value of variable d at time t.
$T,\ D$	Window length and dimension	T is the retained history length and test-start index, D is the number of variables.
X_{obs}	Incomplete observed series	Available historical input window.
$M \in \{0,1\}^{T\times D}$	Observation mask	$M_{t,d} = 1$ observed, $M_{t,d} = 0$ missing.
$\mathcal{G}$	Missing positions	$\mathcal{G} = \{(t,d) \mid M_{t,d} = 0\}$.
$\widetilde{X}$	Initial completion	Component-wise M3 completion serving as the projection starting point.
δ	Correction at missing positions	$\delta_{t,d} = 0,\ \forall (t,d) \notin \mathcal{G}$.
$\widehat{X} = \widetilde{X} + \delta$	Final completion	Preserves observations and supplies a complete TSFM input.

Continued on next page

Table 5: (continued)

Symbol	Meaning	Definition
$\Phi(\cdot) : \mathbb{R}^{T\times D} \to \mathbb{R}^K$	Structural proxy	$K = 6$, describes trend, seasonality, residual dependence, and spectral complexity.
$\phi_{\text{target}} \in \mathbb{R}^K$	Structural target	Target vector estimated by M1 in the effective window.
W_{act}	Effective window	Terminal window of the retained history, entirely before T.
$\omega^{\text{use}}_{t,d}$	Downstream-use importance	Potential forecasting harm from error at (t, d).
$\omega^{\text{lev}}_{t,d}$	Structural leverage	Leverage of (t, d) for changing the structural proxy.
$r_{t,d}$	Positional cost	$r_{t,d} = \omega^{\text{use}}_{t,d}/(\omega^{\text{lev}}_{t,d} + \varepsilon)$.
$v = (v_1, \ldots, v_K)$	Structural weight vector	Relative importance of structural dimensions; uniform by default.
λ	Regularization coefficient	Balances structural discrepancy and positional regularization.
$\eta_{\max},\ \sigma_d$	Amplitude-bound parameters	$\eta_{\max}$ is the base amplitude ratio, σ_d is the scale estimate; the bound tightens with use importance.
ε	Numerical-stability constant	Avoids division by zero in positional costs and normalization.
$\mathcal{C}$	Calibration set	Historical calibration windows for optional model-aware alignment.
$g : \mathbb{R}^{T\times D} \to \mathbb{R}^{H\times D}$	Downstream TSFM	Pretrained, frozen target TSFM.
H	Forecast horizon	Specified by the forecasting task.
$Y \in \mathbb{R}^{H\times D}$	Future ground truth	Available only for evaluation; excluded from imputation.

B.2 M1: Proxy Definitions, Effective Window, and Estimators

Table 6: Six dimensions of the structural proxy.

Aspect	Metric	Definition	Interpretation
Trend	Trend strength TS	$1 - \text{Var}(R)/\text{Var}(T + R)$, T, R denote trend and residual components	Trend strength relative to residual variation
Trend	Trend linearity TL	Linear contribution in an orthogonal linear–quadratic regression of the trend	Orthogonal bases isolate the linear contribution from quadratic curvature
Seasonal	Seasonal strength SS	$1 - \text{Var}(R)/\text{Var}(S + R)$, S denotes the seasonal component	Seasonal strength relative to residual variation
Seasonal	Seasonal correlation SC	Mean pairwise correlation among complete segments of the dominant period	Consistency of seasonal shapes across periods
Residual dependence	Lag-one residual autocorrelation RA	Lag-one autocorrelation of residuals	Linear dependence between adjacent residuals
Spectrum	Spectral entropy SE	Shannon entropy of the normalized power spectrum	Complexity of the spectral distribution

The ideal structural target is $\phi_{\text{target}} = \Phi(X)$, but complete X is unavailable during imputation. Computing $\Phi(X_{\text{fill}})$ from a simple completion introduces dimension-dependent biases under block missingness. For shape descriptors—TS, TL, SS, and SE—a single completion imposes a particular smooth shape. Unstable STL decomposition near gaps can bias TS and SS downward; linear bridging increases the linear share of the trend and can bias TL upward; boundary discontinuities cause spectral leakage and alter SE. For texture descriptors—SC and RA—synthetic bridges or generated texture can create spurious periodic consistency and adjacent correlation. Fewer complete periods further weaken SC estimation.

These biases motivate different estimators. Shape dimensions are averaged over plausible completions to reduce dependence on one assumed gap shape. Texture dimensions instead use visible observations to exclude synthetic imputation artifacts.

The effective window lies at the end of the retained historical prefix, with T denoting the test-start boundary. Its length is determined by forecast horizon H, the number of rolling windows n, and maximum context $C_{\max}$. The test-split proportion and rolling-window limit constrain n; Equation 4 defines the resulting window entirely within $[0, T)$.

The GP is fitted only to standardized observed values, with time normalized to the unit interval. A squared-exponential kernel, a periodic kernel when identifiable, and a noise kernel describe candidate shapes. Posterior samples replace missing entries only, retaining actual observations elsewhere. We average structural measurements over candidates for TS, TL, SS, and SE. SC and RA use the observation-based estimators in Equation 3.

Here S_i is the ith complete period of the observation-based seasonal template; r_t is an observed residual with mean $\bar{r}$; $\mathcal{O}$ contains observed positions; and $\mathcal{P}$ contains adjacent pairs that are both observed. If the period is unidentifiable or valid periods or pairs are insufficient, SC and RA default to zero. Missing segments do not contribute, preventing artificial bridges from inflating seasonal or residual correlation.

$$\phi_{\text{target}} = [\phi_{\text{GP}}^{\text{TS}},\ \phi_{\text{GP}}^{\text{TL}},\ \phi_{\text{GP}}^{\text{SS}},\ \phi_{\text{GP}}^{\text{SE}},\ \phi_{\text{obs}}^{\text{RA}},\ \phi_{\text{obs}}^{\text{SC}}]^{\top}$$

For TS, TL, SS, and SE, GP posterior sampling estimates $\mathbb{E}_{\text{GP}}[\Phi(X) \mid X_{\mathcal{O}}]$. Under the fitted GP and finite second moments, this conditional expectation minimizes squared error for estimating $\Phi(X)$ given observations; the sample mean is its Monte Carlo approximation. It generally differs from $\Phi(\mathbb{E}_{\text{GP}}[X \mid X_{\mathcal{O}}])$, retaining posterior uncertainty through the structural calculation. This optimality is conditional on the assumed GP; accuracy on real series still depends on model fit.

For SC and RA, bridging or generated texture can change periodic consistency and adjacent dependence. Observation-based templates and visible residual pairs reduce dependence on the chosen gap completion. Effective sample sizes shrink with missingness, so insufficient periods or pairs trigger conservative defaults to maintain numerical stability.

B.3 M3: Component Recovery and Fallbacks

Temporary completion serves only period detection and decomposition; component values at missing positions are re-estimated using the mask. Single-period series use STL, multiple periods use multi-seasonal decomposition, and unstable periodicity falls back to trend–residual decomposition. Sparse or entirely missing columns receive conservative handling without forced decomposition. For an interior block $[s, e)$, trend and residuals connect the two boundaries:

$$\widehat{T}(t) = T(s-1) + \frac{t-(s-1)}{e-(s-1)}(T(e) - T(s-1)), \qquad t \in [s, e)$$

$$\widehat{R}(t) = R(s-1) + \frac{t-(s-1)}{e-(s-1)}(R(e) - R(s-1)), \qquad t \in [s, e)$$

For endpoint gaps, the trend extends the nearest boundary value, and residuals decay from the visible endpoint toward zero. A phase template and low-order harmonic fit based on observed positions form the seasonal component:

$$\widehat{S} = \alpha_p\, S_{\text{pat}} + (1 - \alpha_p)\, S_{\text{harm}}$$

The weight α_p increases with observed phase coverage, and harmonic order is limited by the period length. The seasonal component is zero when the period is unidentifiable or fewer than one period of observations are available. The initialization is $\widetilde{X}_d = \widetilde{T}_d + \widetilde{S}_d + \widetilde{R}_d$. At imputed positions, M3 also supplies a diagnostic trend confidence band, available as an optional shape constraint:

$$\text{band}(t,d) = c\,\sigma_d^{\text{obs}}\sqrt{\Delta(t,d)+1}$$

Here $\Delta(t,d)$ is the distance to the nearest observed point, σ_d^{obs} is the scale of observed segments, and c is a constant. The band widens with distance from observations; it does not alter the default M4 constraints. Trend, seasonality, and residuals recover low-frequency shape, periodic phase, and boundary continuity, respectively. Subsequent bounded projection cannot readily repair a severely distorted starting point.

B.4 M2 and M4: Sensitivity, Optimization, and Acceptance

Proximity importance decays exponentially with distance from the forecast origin; the decay rate controls its temporal falloff. Local information combines normalized local variance, first-difference magnitude, and a change-point posterior (Adams and MacKay, 2007). The model factor defaults to one. Optional block-level saliency supplies ω^{model} when enabled, as described in Appendix B.5.

The derivative of Φ_k with respect to $x_{t,d}$ measures that position's marginal influence on structural dimension k. A position has high leverage if a small perturbation substantially changes trend strength, seasonal strength, or spectral entropy. We estimate sensitivities by finite differences at sampled missing positions. For each sampled point, a copy of the observation mask marks only that point as visible; both the baseline M3 completion and its perturbed counterpart use this evaluation mask. This preserves the perturbation inside Φ instead of overwriting it through internal interpolation. The original mask remains unchanged. Sensitivities are interpolated at unsampled positions, then robustly normalized and lower-bounded to avoid singularities at zero leverage.

The constant ε provides numerical stability. A large r indicates high downstream cost relative to structural effect and discourages modification; a small r favors correction. Together, ω^{use}, ω^{lev}, and r form the importance maps $\mathcal{I}$ used by M4. Before optimization, each map is robustly quantile-clipped and linearly mapped to the unit interval, aligning scales across datasets and limiting the effect of extreme values.

Locally, the effect of missing-position adjustments on structural proxies can be approximated linearly around the current complete window. Greater structural leverage therefore implies greater structural change for a given adjustment magnitude. Positional regularization must account for both risk and structural effect: use importance alone can overprotect downstream-relevant positions, while leverage alone can allocate changes to forecast-sensitive positions. Their ratio prices downstream risk relative to structural effect, favoring positions that can efficiently alter structure at lower relative forecasting risk.

Each block basis contains a constant, a ramp, and first-order sine and cosine terms. A Hann envelope smoothly attenuates changes near block boundaries. Typical blocks have four parameters and are coupled through a common window-level structural target. M-plus is only an evaluation mask for optimization; it does not turn generated values into genuine observations or change the information available to M1.

L-BFGS-B optimization minimizes J over block parameters θ. Each evaluation assembles $\widetilde{X} + \delta$ and computes structural loss and regularizers in the effective window. Gradients are approximated by finite differences in parameter space. Parameter dimension grows with the number of gaps, while structural evaluation also depends on window length. Initialization at $\theta = 0$ gives $\delta = 0$, corresponding to M3. Amplitude constraints are enforced in two stages: L_{bound} penalizes violations during optimization, then pointwise hard clipping enforces the bounds after optimization. Observations are restored before checking the clipped candidate's structural loss.

Final acceptance recomputes the candidate's structural loss using the same M-plus mask, period, and weights as optimization. The candidate is accepted only if finite and its structural loss does not exceed that of $\widetilde{X}$ within tolerance; otherwise the method returns $\widetilde{X}$. This is an estimated-target structural acceptance test; clean-reference fidelity and downstream forecast preservation are evaluated separately in Appendices E.5 and E.6.

Selective activation compares M3 and projection on two pseudo-missing blocks drawn from observed history per input. Full projection requires a calibration recovery gain strictly greater than 0.25%; unavailable calibration or insufficient information returns M3. This separates M3's numerical recovery

from M1/M2/M4 refinement: the latter seeks higher structural fidelity while limiting disturbance of the existing completion through positional penalties, amplitude bounds, gating, and fallback.

Projection denotes constrained structural correction of $\widetilde{X}$ under observation consistency and amplitude bounds. The objective balances weighted structural discrepancy against positional costs. It requires neither exact target matching nor the geometry of orthogonal projection. Low-dimensional block parameterization limits unnecessary high-frequency variation in candidate corrections; final amplitude clipping and structural-loss acceptance determine the output.

B.5 M5: Structural Weights, Optional Saliency, and Test-time Alignment

M5 first estimates which structural dimensions matter most to the target TSFM. On complete calibration windows entirely before T, a controlled perturbation π_k of dimension k yields the average forecast-loss increment

$$\Delta_k = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} [\mathcal{L}(g(\pi_k(h_i)), y_i) - \mathcal{L}(g(h_i), y_i)]$$

Here $\mathcal{C}$ is the calibration set, h_i and y_i are histories and targets, and g is the target model. Equation 12 clips negative increments to zero and applies temperature τ, yielding nonnegative weights that sum to one.

When M5 is enabled, w^{model} replaces the default uniform six-dimensional weights in M2 and M4, directing projection and positional costs toward model-sensitive dimensions.

Block-level saliency compares two forecasts with the same initialized context, forecast origin, and inference settings. The occluded version replaces only the block's imputed values with the mean of visible pre-gap neighbors:

$$\text{sal}_b = \text{mean}| \, g(\widetilde{X}_{[:e]}) - g(\widetilde{X}^{(b)}_{[:e]}) \, |$$

Saliency is robustly normalized and injected into ω^{model}; insufficient pre-gap observations skip this step. This interface was not enabled in the M5 experiments. Test-time alignment instead uses actual visible post-gap observations before T to compute clipped, distance-weighted forecast residuals:

$$u = \frac{\sum_t e^{-d_t/\ell} \, e_t}{\sum_t e^{-d_t/\ell}}, \qquad \widehat{X}_b \leftarrow \widehat{X}_b + \eta \, u$$

Here $e_t = \text{clip}(y_t - \widehat{y}_t, \pm c\sigma)$ is an observable residual, d_t is distance from the block, ℓ is a decay scale, and η is a step size. Feedback uses only actual visible post-gap values, not missing ground truth or downstream evaluation labels. Insufficient information retains the M4 result. M5's benefits are model-dependent, and fallback to the model-unaware version remains available. The main experiments therefore disable M5 and evaluate its effects separately.

B.6 Computation, Extensions, and Conservative Fallbacks

M1's main costs are window-local GP fitting and structural calculations; M2 requires sensitivity estimation; M3 uses period detection, local statistics, and gap traversal, typically costing less than GP fitting or TSFM inference; and M4 iteratively optimizes a low-dimensional perturbation. Selective activation executes full projection only when offline calibration supports it, avoiding full projection computation elsewhere. Optional M5 adds model calibration and block-level queries. Without M5, computation mainly involves window-local statistics and low-dimensional optimization, with no dedicated imputation-model training.

The modules permit scenario-dependent extensions. A pool of historical windows at matching frequency can supply donor seasonal templates to M3. Available TSFM interfaces can provide positional importance or sensitivity estimates to M2 or M5. Short or nonperiodic series use trend–residual decomposition and basic reconstruction. Additional proxy dimensions require correspond-

ing changes to M1 targets, M2 structural weights, and M4 discrepancies. SCOPE thus also provides an extensible framework for TSFM input repair.

Conservative fallbacks accompany external information throughout the pipeline. M1 uses only observations for texture dimensions and conservative window-level estimates when fitting degenerates. M2 robustly clips and normalizes importance maps. M3 handles sparsely observed columns conservatively. M4 combines amplitude bounds with final structural acceptance, returning M3 when acceptance fails, while selective activation filters unsupported scenarios. M5 is disabled when unavailable or insufficiently beneficial. These mechanisms permit more conservative behavior under high missingness, unstable periodicity, or inaccessible models.

C Data, Models, and Evaluation Protocol

All methods share missingness masks, rolling origins, and target TSFMs. Complete future labels are stored separately. Tables 7–9 specify the datasets, models, and candidate methods.

C.1 Datasets and Preprocessing

Azure2019 records cloud VM CPU utilization and belongs to the CloudOPS domain (Qiao et al., 2026).

Series are sorted by timestamp and checked for sampling intervals. Historical and forecast windows must come from the same contiguous segment. Imputation normalization uses only visible values of the retained incomplete history before T; forecasting follows each TSFM's prescribed input interface. Since some TSFMs natively support only univariate inputs, each variable of a multivariate dataset is treated as an independent series, ensuring a common input format across models.

Table 7: Evaluation datasets: 27 public time series.

Dataset	Domain	Frequency	Time steps	Variables	Period
Australia_Solar_H	Energy	Hourly	35064	3	24
Coastal_T_S_15T	Nature	15 min	23134	3	96
Coastal_T_S_20T	Nature	20 min	8198	1	72
Coastal_T_S_5T	Nature	5 min	105408	3	288
Coastal_T_S_H	Nature	Hourly	8784	3	24
ETTh1	Energy	Hourly	17420	7	24
ETTh2	Energy	Hourly	17420	7	24
ETTm1	Energy	15 min	69680	7	96
ETTm2	Energy	15 min	69680	7	96
EWELD_Load_15T	Energy	15 min	20544	1	96
Finland_Traffic_15T	Transport	15 min	35136	1	96
NE_China_Wind_H	Energy	Hourly	8764	1	24
OpenElectricity_NEM_5T	Energy	5 min	43488	1	288
Port_Activity_D	Transport	Daily	2127	2	7
Supply_Chain_Customer_D	Transport	Daily	2007	1	7
Supply_Chain_Location_D	Transport	Daily	2007	1	7
Water_Quality_Darwin_15T	Nature	15 min	18985	6	96
azure2019_U_5T	CloudOPS	5 min	2006	3	288
current_velocity_15T	Nature	15 min	17953	6	96
current_velocity_20T	Nature	20 min	16759	6	72
current_velocity_5T	Nature	5 min	26486	1	288
current_velocity_H	Nature	Hourly	5587	6	24
electricity	Energy	Hourly	26304	321	24
exchange_rate	Econ/Fin	Daily	7588	8	7

Continued on next page

Table 7: (continued)

Dataset	Domain	Frequency	Time steps	Variables	Period
national_illness	Healthcare	Weekly	966	7	52
traffic	Transport	Hourly	17544	862	24
weather	Nature	10 min	52696	21	144

C.2 Frozen TSFMs and Candidate Imputers

All five models retain frozen pretrained parameters without additional training on the evaluation datasets. The first four backbones serve the model-unaware experiments, and VisionTS++ is additionally used in the M5 study.

Table 8: Time-series foundation models used in evaluation.

Model	Parameters	Context cap	Architecture	Hugging Face repository
Sundial	128 M	2880	AR-based LLM for TS	thuml/sundial-base-128m
Chronos-2	120 M	8192	Encoder-only (T5-inspired)	amazon/chronos-2
TimesFM-2.5	200 M	4096	Decoder-only	google/timesfm-2.5-200m-pytorch
Kairos	23 M	2048	Encoder–decoder Transformer	mldi-lab/Kairos_23m
VisionTS++	Undisclosed	4000	Vision-style TS encoder	Lefei/VisionTSpp

The listed `amazon/chronos-2` checkpoint is a 120M-parameter, encoder-only model inspired by the T5 encoder and produces multi-step quantile forecasts (Ansari et al., 2025), as described in its official model card.[1]

Table 9: Candidate imputation methods and categories.

Method	Category	Description
Zero	Simple	Fill missing entries with 0
Mean	Simple	Impute with each column's mean
Forward	Simple	Carry the last observation forward
Linear	Simple	Linear interpolation between adjacent observations
Kalman-Struct	State space	Local-linear-trend Kalman smoothing
Kalman-ARIMA	State space	Kalman smoothing in AR(p) state-space form (approximate ARIMA)
STL-Kalman	State space	STL decomposition followed by Kalman smoothing of residuals
GP-RBF	Machine learning	Univariate GP regression with an RBF kernel
CSDI	Deep learning	Conditional score-based diffusion imputation (NeurIPS 2021)
ImputeFormer	Deep learning	Low-rank-constrained Transformer imputation (KDD 2024)
TimeMixer	Deep learning	Multiscale decomposable temporal mixing (ICLR 2024)
TimesNet	Deep learning	General temporal network with 2-D periodic transformations (ICLR 2023)

Continued on next page

[1] https://huggingface.co/amazon/chronos-2

Table 9: (continued)

Method	Category	Description
BRITS	Deep learning	Bidirectional recurrent imputation (NeurIPS 2018)
SCOPE-mu (ours)	Structure- and importance-aware projection	Default SCOPE configuration (M1–M4)
SCOPE-ma (ours)	Structure- and model-aware projection	SCOPE-mu with optional model-aware alignment

C.3 Offline Imputation, Rolling Forecasts, and Aggregation

Let $X^{\text{full}}[0 : T_{\text{full}}]$ denote the original series and T the start of its terminal held-out test segment. Before any imputation preprocessing, we remove $X^{\text{full}}[T : T_{\text{full}}]$ and retain only the incomplete historical prefix $X_{\text{obs}}[0 : T]$. Indices use half-open intervals. GIFT-style rolling evaluation (Aksu et al., 2024) uses origins $o_j \geq T$ and scores predictions against $X^{\text{full}}[o_j : o_j + H]$. Thus the global truncation boundary is no later than any evaluated origin; it is applied before normalization and imputation, rather than only when constructing TSFM inputs.

All imputation-side data access is restricted to visible entries with $t < T$: normalization; period detection and M3 decomposition; M1 GP fitting and texture statistics; M2 importance maps; and M4 structural evaluation. Pseudo-missing gating uses retained historical observations, and both the history and target of an M5 calibration pair lie before T. M5 post-gap feedback likewise uses only visible observations before T. A point to the right of a historical gap is therefore usable only within this retained prefix. Equation 4 uses this truncated history length T; $nH + C_{\max}$ determines window size, not permission to read test targets.

Indexed data-access procedure.

1. Split the series at T; retain test targets separately for scoring. Give the imputer only $X_{\text{obs}}[0 : T]$, its mask, and task-index metadata.
2. Estimate normalization, structural quantities, and calibration quantities from the retained historical observations; compute $\widehat{X}[0 : T]$ and keep this completion fixed during rolling evaluation.
3. For each origin $o_j \geq T$, construct the TSFM context from the imputed history and information available strictly before o_j. Any observations available as the origin advances remain subject to this time boundary and do not re-enter the fixed imputation or calibration.
4. Produce the forecast, then access $X^{\text{full}}[o_j : o_j + H]$ to compute errors. Scoring does not update normalization, structural targets, gating, or M5.

Consequently, the latest observation accessible to any imputation or calibration component is $T-1 < o_j$, while the TSFM context ends at $o_j - 1$. Complete historical values hidden by the missingness mask are used only for offline evaluation, including structural fidelity. Evaluation records include dataset, variable, time origin, history length, horizon, missingness, random seed, imputer, TSFM, and inference configuration for stratified aggregation.

Contiguous missing blocks represent interruptions such as sensor downtime, communication outages, or batches of lost packets. Compared with independent missing points, they more severely disrupt local dynamics, periodic phase, and trend continuation. Missing rates $\{10\%, 20\%, 30\%\}$ and block lengths $\{50, 100, 150\}$ define the nine main settings.

MAE, MSE, and sMAPE evaluate future forecasts. MAE and sMAPE use the 0.5 quantile; MSE compares the model-interface point forecast with the target. For Chronos-2, the official `predict_quantiles` interface returns its 0.5 quantile in the `mean` field.[2] Accordingly, "point forecast" refers to the actual returned quantity. MAE is primary, and MSE and sMAPE supplement the

[2]https://github.com/amazon-science/chronos-forecasting/blob/main/src/chronos/chronos2/pipeline.py

general method comparisons. The native/default comparison reports MAE only. Structural fidelity is evaluated separately against complete historical references.

Dataset-level tables average raw MAE within each comparison group, retaining the original evaluation records for Baseline-5 and Reference-9. The same method can have different group means; comparisons use records from the same group. Reported percentage MAE reductions average paired relative gains across configurations. Cross-dataset summaries average configuration-wise relative errors with the stated reference set to 100: SCOPE-mu for Figures 5(a) and 11, Linear for scenario tables, and native/default handling for Table 2. The configuration is a (TSFM, dataset, missingness setting) triple. Ranks, paired outcomes, and counts of outperforming complete inputs provide complementary comparisons.

Baseline-5 comprises CSDI (Tashiro et al., 2021), ImputeFormer (Nie et al., 2024), TimeMixer (Wang et al., 2024), TimesNet (Wu et al., 2023), and BRITS (Cao et al., 2018), compared under equal budgets and full budgets with additional training epochs. Reference-9 comprises SCOPE-mu, Linear, Forward, Mean, Zero, STL-Kalman, Kalman-Struct, Kalman-ARIMA, and GP-RBF. Reference-9 uses shared random seeds, and Kairos is reset to its initial inference state before each complete evaluation.

The full-budget regime increases training epochs beyond the equal-budget regime, explicitly testing sensitivity to additional baseline training. ImputeFormer's normalized MAE improves from 123.2 to 113.0, CSDI's from 133.7 to 129.6, and BRITS's from 161.6 to 154.9. SCOPE retains the best combined rank (1.73 and 1.83), so the reported ordering is not confined to the smaller training budget. These are comparisons of the evaluated implementations under two budgets, not claims of exhaustive tuning or convergence. Timing includes deep methods' per-input training and excludes SCOPE's precomputed gate calibration, as specified below.

C.4 Timing Scope and Statistical Denominators

Tables 13 and 16 use first-execution records for 81 configurations: Kairos-23M, 27 datasets, three missing rates, and block length 50. Subsequent cache hits are excluded. Timing includes imputation, per-input training for deep methods, Kairos forecasting, metric computation, and I/O. It excludes TSFM preloading and precomputed SCOPE gating calibration. Relative costs use SCOPE-mu as the reference. System-load differences across runs can affect individual measurements.

The main study comprises 972 model configurations; timing fixes Kairos-23M and block length 50 (81 configurations). Each M5 backbone has 243 configurations. The original local-neighborhood fidelity comparison uses 81 dataset–missing-rate pairs. The new M3–SCOPE structural and gate comparisons use 243 inputs, while matched ablation and forecast non-inferiority fix block length 50 (81 inputs, 324 model configurations). Forecast adjustment magnitude uses all 972 model configurations. Dataset-cluster bootstrap intervals use 10,000 draws and preserve within-dataset dependence.

C.5 Experimental Parameters and Reproducibility

Table 10: Experimental settings. Gate calibration uses observations in the retained historical prefix only.

Parameter	**Setting**
GP posterior samples / training-point cap	20 / 600
Amplitude bound $\eta_{\max}$	0.2
Penalty weights $\lambda_{\text{pos}}/\lambda_{\text{bound}}$	1 / 10
Structural-loss constant c	0.1
Proximity decay rate	3.0
M2 finite-difference step	0.01 in the standardized domain
Optimizer	L-BFGS-B; at most 30 iterations
Optimizer tolerances	ftol $= 10^{-5}$; gtol $= 10^{-6}$

Continued on next page

Table 10: (continued)

Parameter	Setting
Gate calibration	2 pseudo-missing blocks per input; recovery gain strictly greater than 0.25%
Forecast horizon H	48 for minute/hour; 30 for daily; 8 for weekly
Rolling-window count / stride	20 / H

The numerical settings in Table 10 apply to the reported experiments. Normalization, period estimation, GP fitting, and the two-block gate calibration use only visible entries before T. Model weights remain frozen. The context limits in Table 8 specify the reported context caps; TimesFM-2.5 uses 4096 in these experiments, although the released model supports up to 16k (Google Research, 2025). Actual histories can be shorter. The code and original data for this work are submitted in the Supplementary Material.

D Full Forecasting Results and Execution Costs

Tables 11, 14, and 17 report raw MAE averaged over 36 model–rate–block configurations per dataset. Row multipliers affect display only; bold marks the lowest displayed value. Cross-dataset summaries retain configuration-wise normalization.

D.1 Baseline-5: Equal Budget

Table 11: Dataset-level raw MAE for equal-budget Baseline-5 (lower is better). MAE equals the displayed value multiplied by the row multiplier.

Dataset	Factor	SCOPE-mu	CSDI	Impute-Former	TimeMixer	TimesNet	BRITS
Australia_Solar_H	10^{5}	1.515	1.464	**1.305**	1.877	1.848	1.840
Coastal_T_S_15T	10^{-2}	**5.319**	5.380	5.885	6.344	6.345	6.494
Coastal_T_S_20T	10^{-1}	**1.045**	3.462	2.799	5.314	5.004	5.334
Coastal_T_S_5T	10^{1}	4.479	4.486	4.501	4.485	**4.478**	4.492
Coastal_T_S_H	1	**5.642**	11.734	6.226	10.686	10.456	10.611
ETTh1	1	**1.197**	1.530	1.552	1.821	1.833	2.019
ETTh2	1	**2.956**	3.299	3.117	3.754	3.742	3.493
ETTm1	10^{-1}	**7.935**	8.698	8.951	12.599	12.473	11.026
ETTm2	1	**2.185**	2.620	3.130	4.139	4.093	3.722
EWELD_Load_15T	10^{1}	**3.427**	4.717	5.329	5.332	5.347	5.372
Finland_Traffic_15T	10^{1}	**3.204**	3.829	4.132	4.083	4.031	3.940
NE_China_Wind_H	10^{-1}	**4.187**	5.265	4.785	5.488	5.512	5.667
OpenElectricity_NEM_5T	10^{2}	**2.134**	2.768	2.703	2.964	2.959	2.896
Port_Activity_D	10^{3}	2.766	2.927	**2.732**	2.779	2.859	2.935
Supply_Chain_Customer_D	10^{4}	**2.545**	2.842	2.675	2.715	2.686	2.772
Supply_Chain_Location_D	10^{4}	**1.270**	1.440	1.301	1.312	1.305	1.340
Water_Quality_Darwin_15T	10^{-1}	**9.190**	9.886	10.089	25.829	25.737	22.508
azure2019_U_5T	10^{1}	**1.322**	1.613	1.398	1.547	1.615	1.611
current_velocity_15T	10^{-2}	**9.033**	17.323	10.437	23.977	22.430	24.424
current_velocity_20T	1	**3.937**	4.356	4.507	4.583	4.580	4.713
current_velocity_5T	10^{-2}	**5.497**	11.375	10.550	10.342	10.511	10.572
current_velocity_H	1	**4.788**	5.052	4.977	5.153	5.205	5.235
electricity	10^{2}	1.182	**1.160**	1.203	1.382	1.350	1.342
exchange_rate	10^{-2}	**1.261**	1.922	1.443	1.748	1.712	2.151
national_illness	10^{5}	**1.034**	1.552	1.103	1.155	1.136	1.143
traffic	10^{-3}	**4.775**	5.325	4.782	6.832	6.744	6.800
weather	1	**2.798**	3.013	4.059	4.436	4.317	6.521

Table 12: Average ranks for equal-budget Baseline-5.

Rank	Method	MAE	MSE	sMAPE	Combined
1	**SCOPE-mu**	**1.59**	**1.82**	**1.78**	**1.73**
2	ImputeFormer	2.73	2.96	2.75	2.81
3	CSDI	3.21	3.14	3.25	3.20
4	TimesNet	4.32	4.16	4.24	4.24
5	TimeMixer	4.53	4.41	4.52	4.49
6	BRITS	4.62	4.51	4.47	4.53

Table 13: First-execution costs for equal-budget Baseline-5 (81 configurations).

Method	Mean time (s/eval)	Total time (s)	Relative to SCOPE
SCOPE-mu	4.02	325.6	1.00×
TimeMixer	2.07	168.0	0.52×
ImputeFormer	3.12	252.4	0.78×
TimesNet	4.38	354.8	1.09×
CSDI	5.28	427.8	1.31×
BRITS	37.14	3008.1	9.24×

SCOPE-mu's configuration-wise MAE win rates against CSDI, ImputeFormer, TimeMixer, TimesNet, and BRITS are 80.8%, 81.1%, 92.8%, 91.5%, and 95.1%, respectively. First-execution time does not represent each method's full lifecycle cost; Appendix C.4 defines its scope.

D.2 Baseline-5: Full Budget

Table 14: Dataset-level raw MAE for full-budget Baseline-5 (lower is better). MAE equals the displayed value multiplied by the row multiplier.

Dataset	Factor	SCOPE-mu	CSDI	Impute-Former	TimeMixer	TimesNet	BRITS
Australia_Solar_H	10^{5}	1.515	1.380	**1.319**	1.887	1.872	1.853
Coastal_T_S_15T	10^{-2}	5.319	**5.317**	5.636	6.337	6.359	6.260
Coastal_T_S_20T	10^{-1}	**1.045**	4.621	1.086	5.307	5.093	4.488
Coastal_T_S_5T	10^{1}	4.479	4.484	4.480	4.489	**4.465**	4.492
Coastal_T_S_H	1	**5.642**	6.461	5.906	10.715	10.212	10.807
ETTh1	1	**1.197**	1.286	1.464	1.823	1.806	1.809
ETTh2	1	**2.956**	3.138	3.182	3.755	3.752	3.675
ETTm1	10^{-1}	**7.935**	8.430	8.393	12.619	12.560	11.834
ETTm2	1	**2.185**	2.779	3.244	4.128	4.135	4.013
EWELD_Load_15T	10^{1}	**3.427**	4.967	5.407	5.369	5.302	5.308
Finland_Traffic_15T	10^{1}	**3.204**	3.875	4.094	4.100	4.068	4.102
NE_China_Wind_H	10^{-1}	**4.187**	5.160	4.659	5.520	5.505	5.406
OpenElectricity_NEM_5T	10^{2}	**2.134**	2.554	2.853	2.969	2.972	3.024
Port_Activity_D	10^{3}	2.766	2.896	**2.743**	2.850	2.872	2.797
Supply_Chain_Customer_D	10^{4}	**2.545**	2.733	2.696	2.703	2.705	2.677
Supply_Chain_Location_D	10^{4}	**1.270**	1.281	1.302	1.296	1.299	1.299
Water_Quality_Darwin_15T	10^{-1}	**9.190**	10.778	9.745	25.581	25.506	25.914
azure2019_U_5T	10^{1}	1.322	1.851	**1.308**	1.608	1.567	1.607
current_velocity_15T	10^{-2}	**9.033**	10.758	9.696	25.396	23.059	24.276
current_velocity_20T	1	**3.937**	4.444	4.437	4.579	4.589	4.901
current_velocity_5T	10^{-2}	**5.497**	8.363	8.252	10.575	10.610	9.903
current_velocity_H	1	4.788	5.199	**4.771**	5.209	5.192	5.180
electricity	10^{2}	1.182	**1.107**	1.190	1.384	1.374	1.324

Continued on next page

Table 14: (continued)

Dataset	Factor	SCOPE-mu	CSDI	Impute-Former	TimeMixer	TimesNet	BRITS
exchange_rate	10^{-2}	**1.261**	2.493	1.419	1.751	1.758	1.570
national_illness	10^{5}	**1.034**	1.297	1.056	1.137	1.149	1.162
traffic	10^{-3}	4.775	**4.321**	4.502	6.818	6.868	6.682
weather	1	**2.798**	3.096	3.549	4.388	4.202	4.050

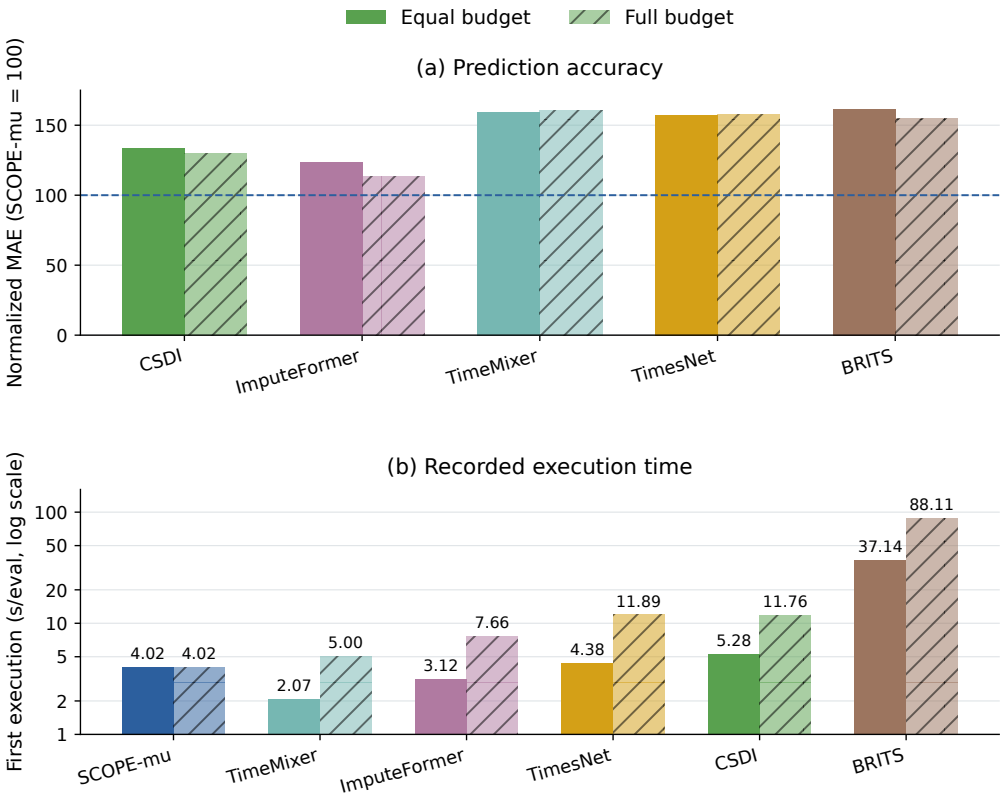

Figure 5: Deep baselines under two training budgets: (a) normalized MAE (SCOPE-mu = 100); (b) first-execution time (log scale; labels in s/eval).

Table 15: Average ranks for full-budget Baseline-5.

Rank	Method	MAE	MSE	sMAPE	Combined
1	**SCOPE-mu**	**1.69**	**1.96**	**1.86**	**1.83**
2	ImputeFormer	2.72	2.75	2.79	2.75
3	CSDI	3.01	2.97	2.91	2.96
4	BRITS	4.25	4.30	4.32	4.29
5	TimesNet	4.56	4.45	4.44	4.48
6	TimeMixer	4.77	4.57	4.69	4.68

Table 16: First-execution costs for full-budget Baseline-5 (81 configurations).

Method	Mean time (s/eval)	Total time (s)	Configurations	Relative to SCOPE
SCOPE-mu	4.02	325.6	81	1.00×
TimeMixer	5.00	405.4	81	1.25×
ImputeFormer	7.66	620.5	81	1.91×
CSDI	11.76	952.7	81	2.93×
TimesNet	11.89	962.7	81	2.96×
BRITS	88.11	7137.1	81	21.92×

Figure 5(a) shows different responses to increased training. ImputeFormer's normalized MAE falls from 123.2 to 113.0, CSDI's from 133.7 to 129.6, and BRITS's from 161.6 to 154.9. TimeMixer and TimesNet change little, from 159.4 and 157.2 to 160.4 and 157.8, respectively. Table 16 and Figure 5(b) report full-budget mean times of 5.00, 7.66, 11.76, 11.89, and 88.11 seconds per evaluation for TimeMixer, ImputeFormer, CSDI, TimesNet, and BRITS: 1.25, 1.91, 2.93, 2.96, and 21.92 times SCOPE-mu's cost. SCOPE-mu retains the same configuration and takes 4.02 seconds per evaluation. Additional training improves some deep methods at higher execution cost; SCOPE-mu retains favorable ranks and an accuracy–cost balance in this comparison.

D.3 Reference-9: Classical Methods

Table 17: Dataset-level raw MAE for Reference-9 (lower is better). MAE equals the displayed value multiplied by the row multiplier.

Dataset	Factor	SCOPE-mu	Linear	STL-Kalman	GP-RBF	Kalman-Struct	Forward	Kalman-ARIMA	Mean	Zero
Australia_Solar_H	10^5	**1.533**	1.965	3.359	2.022	4.877	2.040	1.933	1.968	1.987
Coastal_T_S_15T	10^{-2}	5.233	5.233	6.025	**5.226**	5.992	5.386	5.760	5.951	615.088
Coastal_T_S_20T	10^{-1}	**1.107**	**1.107**	1.246	1.134	1.370	1.196	6.579	6.660	36.185
Coastal_T_S_5T	10^{1}	4.497	4.496	5.431	**4.478**	5.312	4.502	4.483	4.483	4.739
Coastal_T_S_H	1	5.757	5.778	7.056	**5.675**	7.264	6.134	7.081	11.676	10.671
ETTh1	1	**1.263**	1.281	1.402	1.288	1.555	1.412	1.409	1.786	2.640
ETTh2	1	**3.068**	3.336	3.908	3.283	4.513	3.589	3.953	3.960	9.198
ETTm1	10^{-1}	**8.072**	8.374	8.752	8.899	9.470	9.227	14.180	15.173	18.889
ETTm2	1	**2.166**	2.368	2.250	2.558	2.663	2.725	3.743	3.804	7.841
EWELD_Load_15T	10^{1}	**3.653**	5.676	4.542	5.999	6.542	6.267	5.249	5.516	6.209
Finland_Traffic_15T	10^{1}	3.125	3.778	**2.968**	3.894	3.805	4.175	3.843	3.940	4.231
NE_China_Wind_H	10^{-1}	4.466	4.407	**4.402**	4.570	4.577	4.869	4.987	5.062	5.161
OpenElectricity_NEM_5T	10^{2}	**2.300**	2.623	2.638	2.829	2.476	3.481	2.941	2.919	14.411
Port_Activity_D	10^{3}	2.772	2.771	3.533	3.904	3.693	2.857	2.846	2.858	**2.662**
Supply_Chain_Customer_D	10^{4}	**2.632**	2.843	3.017	3.943	3.488	3.287	2.800	2.813	3.478
Supply_Chain_Location_D	10^{4}	**1.267**	1.349	1.510	1.401	1.519	1.406	1.287	1.290	1.504
Water_Quality_Darwin_15T	10^{-1}	9.482	**9.450**	9.490	9.671	11.385	9.945	23.611	25.947	297.469
azure2019_U_5T	10^{1}	1.305	1.306	1.866	1.911	1.515	**1.301**	1.533	1.567	1.391
current_velocity_15T	10^{-2}	**9.212**	**9.212**	10.313	9.302	9.537	10.149	23.182	23.231	194.833
current_velocity_20T	1	**3.903**	4.513	4.315	4.444	5.038	4.732	4.501	4.653	33.817
current_velocity_5T	10^{-2}	8.196	8.350	**6.019**	10.680	7.111	12.180	9.658	9.875	1963.184
current_velocity_H	1	**4.867**	5.647	7.885	5.609	10.222	5.819	5.253	5.265	41.889
electricity	10^{2}	**1.178**	1.414	1.940	1.346	2.753	1.381	1.515	1.539	5.974
exchange_rate	10^{-2}	**1.214**	1.216	1.286	1.271	1.430	1.815	1.246	1.615	15.763
national_illness	10^{5}	1.053	1.053	**1.038**	1.133	1.072	1.059	1.055	1.093	1.322
traffic	10^{-3}	**4.864**	7.642	10.757	7.227	17.727	8.208	6.931	6.980	9.255
weather	1	2.889	**2.881**	**2.881**	2.965	3.152	3.194	3.732	4.422	71.302

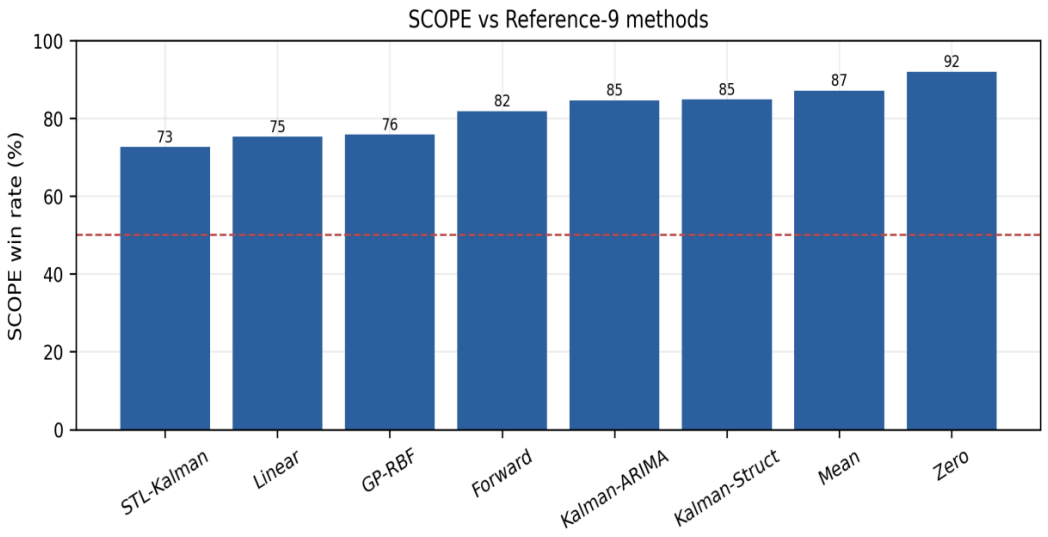

Figure 6: Configuration-wise MAE win rates of SCOPE-mu against the eight Reference-9 comparators, excluding ties.

Table 18: Average ranks for Reference-9 (ties receive the minimum rank).

Rank	Method	MAE	MSE	sMAPE	Combined
1	**SCOPE-mu**	**2.38**	**2.52**	**2.49**	**2.46**
2	Linear	3.55	3.65	3.48	3.56
3	GP-RBF	4.26	4.13	4.09	4.16
4	STL-Kalman	4.49	4.53	4.40	4.47
5	Kalman-ARIMA	5.10	4.67	5.07	4.95
6	Forward	5.10	5.35	4.91	5.12
7	Mean	5.88	5.51	5.75	5.71
8	Kalman-Struct	5.88	5.91	5.76	5.85
9	Zero	7.75	8.12	8.44	8.11

Table 18 gives SCOPE-mu average ranks of 2.38, 2.52, and 2.49 for MAE, MSE, and sMAPE, with a combined rank of 2.46, the best in Reference-9. Linear and GP-RBF follow at 3.56 and 4.16. Against Linear, SCOPE-mu records 548 wins, 180 losses, and 244 ties: a 75.3% win rate excluding ties, wins in 56.4% of all configurations, and a 6.97% mean relative MAE reduction. Against STL-Kalman, it records 686 wins, 258 losses, and 28 ties, with a 72.7% win rate excluding ties and a 10.43% mean relative MAE reduction. Ranks describe relative ordering within this nine-method set and should be interpreted alongside ties, win rates, and error reductions. Gains vary across datasets and missingness configurations.

In Figure 6, SCOPE-mu's win rate exceeds 50% against all eight comparators. Rates against STL-Kalman, Linear, and GP-RBF are 72.7%, 75.3%, and 75.8%; those against Forward, Kalman-ARIMA, Kalman-Struct, Mean, and Zero are 81.8%, 84.6%, 84.9%, 87.1%, and 92.0%. Table 17 reports raw dataset-mean MAE; reference methods remain competitive and achieve lower errors on some datasets. SCOPE-mu's advantage is aggregate performance across datasets and models, with specific gains depending on structure and missingness.

D.4 Native and Default Missing-value Handling

For each model–dataset–missingness configuration, native/default MAE is normalized to 100 and SCOPE MAE is expressed on that scale. Scores are averaged equally across 243 configurations per model (nine per dataset). A gain is 100 minus the unrounded normalized score, so positive values

favor SCOPE. Table 2 reports normalized MAE; Table 19 adds medians, dataset-bootstrap intervals, and configuration-wise wins/losses/ties. Only MAE is used for this comparison.

Chronos-2, Kairos, and TimesFM-2.5 native errors are averaged over three inference seeds within each configuration and paired with SCOPE from the complete Reference-9 results. Sundial uses one matched inference seed per side, with the same missing input, target, and environment.

Table 19: MAE gains over native/default handling (%). Wins/losses/ties count configurations; positive gains indicate lower SCOPE error.

Model	Mean gain	Median gain	95% CI of mean gain	W/L/T
Chronos-2	–4.9451	–0.4085	$[-13.9310, 2.2354]$	112/129/2
Kairos-23M	13.2927	8.6876	$[7.7044, 19.6994]$	181/62/0
TimesFM-2.5	15.3706	11.8216	$[10.8927, 20.1344]$	203/31/9
Sundial (Forward)	13.6428	9.3587	$[9.5548, 17.8657]$	204/37/2

The full grid retains two inputs in which no missingness was actually injected. Excluding them leaves 241 configurations per model and preserves all four mean-gain directions. Confidence intervals resample datasets as clusters 10,000 times; Sundial's intervals condition on the paired seed. Each dataset has equal weight, while medians complement means that can be influenced by large relative-error ratios.

Kairos, TimesFM-2.5, and Sundial show positive average MAE reductions with intervals wholly above zero. Chronos-2 has a 4.9451% mean and 0.4085% median relative deficit for SCOPE; its interval spans zero and provides no equivalence conclusion. Native support can exploit a particular model's input conventions. SCOPE instead supplies one model-agnostic completion procedure across frozen backbones, yielding clear benefits on three of the four evaluated models while leaving Chronos-2's native advantage visible.

E Structural Fidelity, Scenarios, and Module Analysis

Structural fidelity tests the repair mechanism; comparisons with complete inputs assess limited denoising effects; scenario groups characterize the distribution of gains; and ablations, paired projection comparisons, and gate analysis separate numerical recovery from structural refinement.

E.1 Paired Structural-fidelity Results

Table 20: Mean structural-proxy errors. Dimensions average over neighborhood samples; Overall averages equally over dataset–missing-rate combinations.

Method	TS	TL	SS	SC	RA	SE	Overall
SCOPE-mu	0.0721	0.3054	0.0398	0.0155	0.0323	0.0342	0.0876
CSDI-full	0.0811	0.4588	0.0572	0.0423	0.1016	0.0451	0.1668
ImputeFormer-full	0.0872	0.4765	0.0614	0.0134	0.0398	0.0447	0.1062
TimeMixer-full	0.1394	0.8404	0.1256	0.0126	0.0767	0.0949	0.1901
TimesNet-full	0.1385	0.8416	0.1237	0.0142	0.0761	0.0945	0.1901
BRITS-full	0.1173	0.7403	0.0888	0.0142	0.0716	0.0680	0.1656

Table 20 reports all six structural errors. Complete histories are offline evaluation references only and are not used in gating or projection. Each proxy column averages over all neighborhood samples; Overall first averages within dataset–missing-rate combinations, then equally across combinations.

Against CSDI-full, ImputeFormer-full, TimeMixer-full, TimesNet-full, and BRITS-full, SCOPE-mu records structural wins/losses of 63/18, 55/26, 73/8, 74/7, and 68/13. Overall structural errors decrease by 47.5%, 17.6%, 53.9%, 53.9%, and 47.1%, respectively. Structural and forecast rankings need not coincide configuration by configuration.

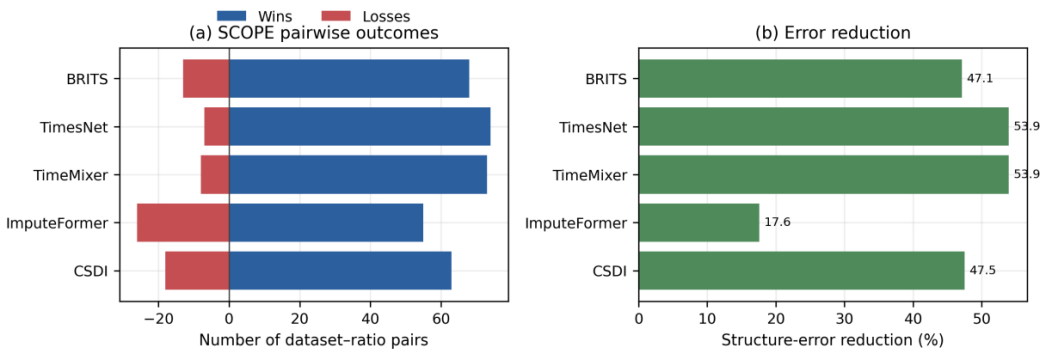

Figure 7: Paired wins, losses, and error reductions for local-window structural fidelity.

E.2 Comparison with Complete Inputs

Across 108 model–dataset pairs from four TSFMs and 27 datasets, we count cases in which imputed inputs outperform complete inputs (clean) on downstream MAE, MSE, and sMAPE. This assesses whether structural repair can produce smoothing or denoising favorable to extrapolation; it does not replace configuration-wise error comparisons.

Table 21: Counts of outperforming complete inputs (clean) across 108 model–dataset pairs.

Method	Pairs	MAE wins	MSE wins	sMAPE wins	Total
SCOPE-mu	108	25	28	27	80
Linear	108	22	26	24	72
STL-Kalman	108	19	22	18	59
CSDI	108	5	11	8	24
ImputeFormer	108	6	9	9	24
TimesNet	108	4	8	10	22
TimeMixer	108	4	9	8	21
BRITS	108	2	5	7	14

In Table 21, SCOPE-mu outperforms clean 25, 28, and 27 times on MAE, MSE, and sMAPE, respectively, totaling 80, the highest count. Linear and STL-Kalman total 72 and 59, while deep baselines total 14–24. The lead over Linear is only eight cases, a smaller advantage than the aggregate ranks might suggest. SCOPE-mu retains some structured denoising capability, but outperforming clean is not common enough to establish that imputed inputs generally improve on complete originals.

E.3 Missing Rates, Block Lengths, and Structural Groups

These groups compare SCOPE-mu, Linear, and five deep baselines. SCOPE-mu and Linear reuse Reference-9 forecasts. MAE is normalized within each configuration to Linear at 100 and averaged equally within groups. Each missing-rate–block-length group contains 108 configurations from four TSFMs and 27 datasets.

Table 22: Normalized MAE by missing rate and block length (Linear = 100; lower is better).

Combination	SCOPE-mu	Linear	CSDI	Impute-Former	TimeMixer	TimesNet	BRITS	Best
10%-50	97.1	100.0	110.1	104.5	126.0	122.6	129.8	SCOPE-mu
10%-100	95.6	100.0	101.1	102.8	112.6	112.2	116.5	SCOPE-mu
10%-150	95.8	100.0	115.6	106.9	117.7	117.3	120.2	SCOPE-mu
20%-50	95.0	100.0	123.8	119.4	150.6	149.9	163.9	SCOPE-mu

Continued on next page

Table 22: (continued)

Combination	SCOPE-mu	Linear	CSDI	Impute-Former	TimeMixer	TimesNet	BRITS	Best
20%-100	90.9	100.0	111.5	119.6	139.0	140.6	147.0	SCOPE-mu
20%-150	90.8	100.0	130.1	130.5	151.8	152.5	161.5	SCOPE-mu
30%-50	91.5	100.0	134.2	123.5	173.9	174.3	173.9	SCOPE-mu
30%-100	89.7	100.0	128.7	128.1	167.1	167.5	167.2	SCOPE-mu
30%-150	91.0	100.0	141.9	146.7	182.4	180.1	194.1	SCOPE-mu

SCOPE-mu has the lowest mean relative MAE among these seven methods in all nine groups, improving on Linear by approximately 2.9%–10.3%. Equal aggregation across groups yields a 6.97% mean reduction, consistent with Section 4.2. Aggregated reductions rise from 3.9% at 10% missingness to 7.8% at 20% and 9.3% at 30%. Block length also matters: at 20% missingness, relative MAE is 95.0, 90.9, and 90.8 for lengths 50, 100, and 150; at 30%, it is 91.5, 89.7, and 91.0, a non-monotonic pattern. Deep baselines also vary across groups. Missing rate and block length should thus be considered jointly.

Structural groups are defined using proxies from complete historical windows for result analysis, and a dataset may belong to multiple groups. Ties in the table use values rounded to one decimal place.

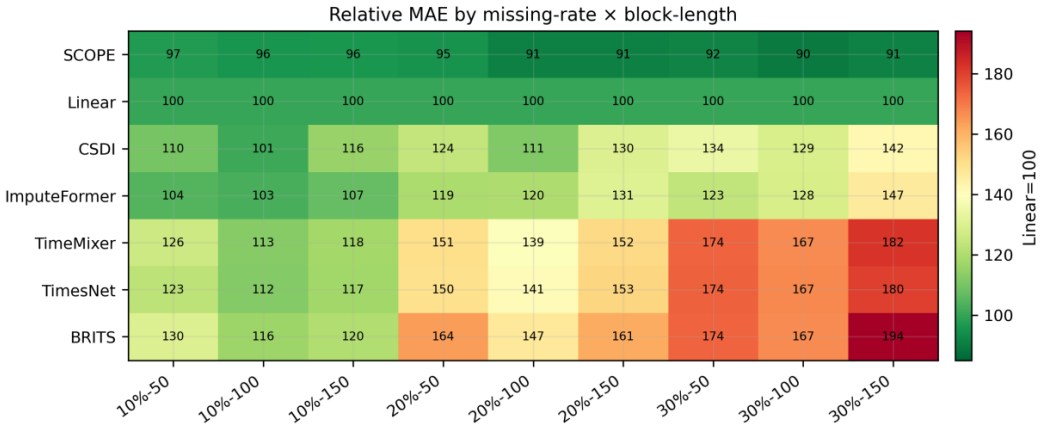

Figure 8: Relative-MAE heatmap for seven methods by missing rate and block length.

Table 23: Normalized MAE by structural group (Linear = 100; lower is better).

Structure type	Datasets	SCOPE-mu	Linear	CSDI	Impute-Former	Time-Mixer	TimesNet	BRITS	Best
Strong Trend	14	96.8	100.0	141.7	138.7	183.5	183.3	192.1	SCOPE-mu
Strong Seasonality	14	86.6	100.0	100.2	101.7	115.1	114.5	116.4	SCOPE-mu
Weak Seasonality	13	100.0	100.0	145.3	140.2	181.0	180.6	191.7	SCOPE-mu / Linear
High Spectral Entropy	14	94.6	100.0	112.6	104.3	115.2	114.2	118.9	SCOPE-mu
High Residual ACF	14	96.7	100.0	137.1	140.5	180.9	181.2	191.4	SCOPE-mu

SCOPE-mu's relative MAE is 96.8, 86.6, 94.6, and 96.7 in the strong-trend, strong-seasonality, high-spectral-entropy, and high-residual-autocorrelation groups. All improve on Linear, with the largest reduction, 13.4%, under strong seasonality. In the weak-seasonality group, both methods round to 100.0, indicating little average difference. CSDI and ImputeFormer also approach Linear under strong seasonality. The results favor SCOPE-mu when seasonal information is clear, while weak-seasonality performance is close to linear interpolation.

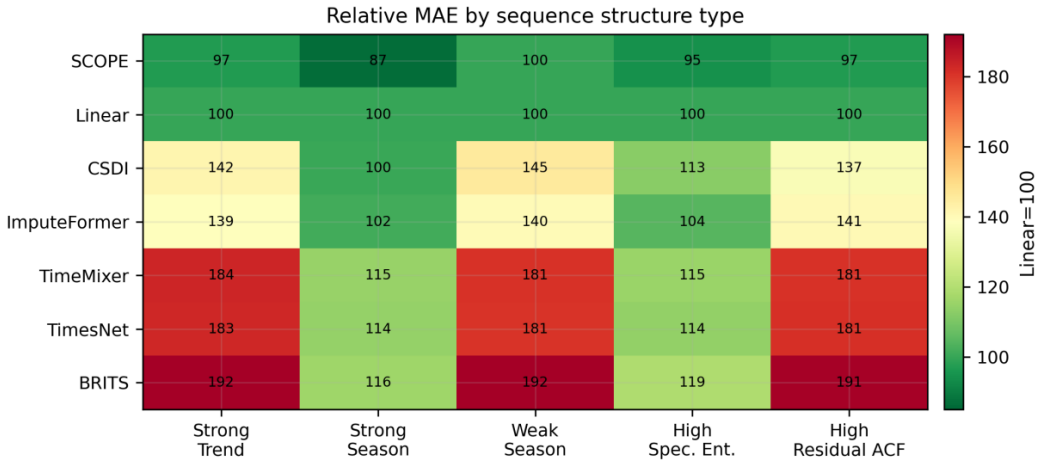

Figure 9: Relative-MAE heatmap for seven methods by structural group.

E.4 Single-module Ablation: Numerical Recovery

The original ablation covers four TSFMs, 27 datasets, and three missing rates at block length 50. Each variant disables or replaces one module. Within each configuration, error change is normalized by the full SCOPE-mu error, then averaged; positive values indicate worse forecasts.

Table 24: Mean paired forecast-error change after single-module ablation (%, block length 50). Positive values indicate higher error than full SCOPE-mu.

Ablation	Variant	MAE change (%)	MSE change (%)	sMAPE change (%)
w/o M1	Disable explicit structure target	+0.37	+0.65	+0.52
w/o M2	Use uniform importance weights	+0.11	+0.18	+0.21
w/o M3	Replace component-wise initialization	+7.24	+16.16	+4.99
w/o M4	Disable structure projection	+0.08	+0.15	+0.21

Removing M3 raises MAE, MSE, and sMAPE by 7.24%, 16.16%, and 4.99%, respectively, the largest changes in Table 24. Removing M1, M2, or M4 also increases all three averages. These effects identify component-wise initialization as the main numerical recovery stage and projection as a smaller refinement. Module effects are conditional on the remaining pipeline and are not additive. The original table retains its original evaluation protocol and full-SCOPE denominator. The matched-seed analysis below instead averages paired seed errors and normalizes gains by each ablated variant; its percentages answer a different comparison and are reported separately.

E.5 Paired Structural Fidelity of Projection

This comparison isolates M3→SCOPE over all 243 inputs, using clean six-dimensional features only as post-hoc references. Both completions use the same observed-data normalization, observation-derived period, and structural evaluation window. For input i, let

$$e_i^{\mathrm{str}}(Z) = \frac{1}{6}\sum_{k=1}^{6} \left|\Phi_k(Z_i) - \Phi_k(X_i^{\mathrm{clean}})\right|, \qquad d_i = e_i^{\mathrm{str}}(\widetilde{X}) - e_i^{\mathrm{str}}(\widehat{X}).$$

We average d_i equally across inputs. The reported relative reduction is the difference of mean errors divided by the M3 mean error, rather than an average of input-wise percentage reductions. Resampling whole datasets 10,000 times preserves their internal missingness configurations.

Table 3 reports mean errors of 0.045734 and 0.043222, an absolute decrease of 0.002512 with 95% interval $[0.000786, 0.004977]$ and a relative reduction of 5.49%. The 97 activated inputs improve

from 0.052680 to 0.046387: an absolute decrease of 0.006293, 95% interval $[0.002150, 0.012619]$, and relative reduction of 11.95%. Input-wise wins/losses/ties are 78/19/146 overall and 78/19/0 among activated inputs. After dataset aggregation they are 20/4/3, with two-sided sign-test $p = 0.0015$ and Wilcoxon $p = 0.0005$.

The improvements principally involve TL, TS, SS, and SE, with small SC changes and a slight deterioration in RA. Thus the supported fidelity result concerns the six-dimensional average. This clean-reference metric also differs from the estimated-target structural loss used in M4 acceptance. Its scale is separate from the neighborhood-based deep-baseline evaluation in Appendix E.1.

E.6 Average Forecast Non-inferiority and Imputation Seeds

The paired comparison fixes block length 50, yielding 81 inputs and 324 model configurations, and uses M3 (the variant without M4) as the comparator. Imputation outputs and gates are held fixed. Each side uses the same inference seeds: one paired seed for Chronos-2, Kairos, and TimesFM-2.5, and five for Sundial. For each configuration, errors are first averaged across these seeds, then the relative MAE gain is computed as

$$g_i = 100\,\frac{\overline{E}_i^{\mathrm{M3}} - \overline{E}_i^{\mathrm{SCOPE}}}{\overline{E}_i^{\mathrm{M3}}}.$$

The prespecified non-inferiority margin permits at most 0.5% relative MAE degradation: the one-sided 95% lower bound for mean gain must exceed -0.5%. A two-sided 90% interval entirely inside $[-0.5\%, 0.5\%]$ additionally meets the mean-equivalence criterion. Intervals use 10,000 dataset-cluster bootstrap draws.

Table 25 reports the corresponding intervals and decisions.

Table 25: Mean forecast non-inferiority and equivalence at block length 50. Gains and intervals are in %; the prespecified non-inferiority margin is 0.5%, with equivalence bounds of $\pm 0.5\%$.

Configurations	Mean MAE gain	90% dataset CI	Non-inferior	Equivalent
All 324	0.0198	$[-0.1578, 0.1787]$	Yes	Yes
Active 120	0.0535	$[-0.4217, 0.4861]$	Yes	Yes

Both all configurations and the activated subset satisfy the average criteria. The full-sample 95% interval is $[-0.1953\%, 0.2102\%]$, consistent with preserved average forecasting performance rather than a statistically confirmed positive forecast gain. The activated group's 90% interval lies close to the equivalence bounds; individual configurations can still worsen.

These intervals condition on the stored imputations and gates. A separate study at 20% missingness, block length 50, 27 datasets, and five imputation seeds finds a Chronos-2 aggregate-MAE standard deviation of 0.2262%, rising to 0.5551% among the 11 activated datasets. It retains one GP numerical-failure fallback under the original implementation. Imputation-side variability therefore remains relevant: the demonstrated non-inferiority applies to the tested fixed imputations and block length, while robustness over new imputation seeds or lengths 100 and 150 requires separate evaluation.

E.7 Gate Activation, Concentration of Effects, and Adjustment Size

The gate tests two pseudo-missing blocks per input and requires a recovery gain strictly greater than 0.25%. Calibration accesses retained historical observations only. Table 26 distinguishes the full grid from the paired block-length-50 subset.

Table 26: Projection-gate activation, with inputs and model configurations counted separately.

Scope	Active inputs	Inactive inputs	Activation	Active model configs
Full grid	97/243	146/243	39.92%	388/972
Paired, length 50	30/81	51/81	37.04%	120/324

The 146 inactive inputs exactly match M3 and account for all 146 structural ties. Refinement is concentrated in the activated group, whose mean structural improvement is larger than the full-grid average (11.95% versus 5.49%). In the paired forecast study, the active group's mean MAE gain of 0.0535% has the same positive direction as the full sample's 0.0198%, and both meet average non-inferiority. This is agreement of aggregate directions; the 19 structural losses among activated inputs and model-dependent forecast effects remain part of the evaluation. The gate is a calibration filter, not a per-input performance guarantee.

Table 27: Mean standardized L1 adjustment over actual forecast contexts (972 model configurations).

Stage	Comparison	Mean adjustment
Initialization	Linear $\rightarrow$ M3	0.0622025
Projection	M3 $\rightarrow$ SCOPE	0.0018401

Adjustment magnitude is standardized by observed-history scales available before each forecast origin, averaged over all positions of the actual TSFM contexts, then equally over windows and the 972 model configurations. Zero changes remain in the average. Linear$\rightarrow$M3 changes the input by 0.0622025 on this scale; M3$\rightarrow$SCOPE changes it by 0.0018401, a ratio of 2.96%. This supports a small numerical refinement around the main recovery stage.

The gate bypasses full projection for 146/243 inputs (60.08%), avoiding their additional optimization calls. Activated corrections use a low-dimensional block basis and at most 30 L-BFGS-B iterations. Together with the small adjustment magnitude, this localizes extra work and limits disturbance of the initialization. The measured 4.02-second execution cost in Appendix C.4 excludes precomputed gating calibration; adjustment size is not a runtime fraction, and a full calibration-inclusive cost is not available from these timing records.

E.8 Matched-seed Module Effects and Inference Variability

For Sundial, the analysis fixes imputed inputs and pairs each M1, M2, or M4 ablation with full SCOPE under five identical inference seeds on 81 block-length-50 configurations. Each configuration uses the ablated variant's five-seed mean MAE as a fixed denominator. We aggregate configuration gains within each seed, then compute the standard deviation (SD) of those five aggregate gains. The Monte Carlo standard error (SE) is $\mathrm{SD}/\sqrt{5}$ and measures uncertainty in the inference mean on this fixed configuration set.

Table 28: Module effects under five matched Sundial inference seeds. The first three numerical columns are in %; SD is the standard deviation of paired aggregate gains and SE is their Monte Carlo standard error.

Module	Mean MAE gain	Paired SD	Mean SE	Effect/SD	Effect/SE
M1	0.2318	0.0234	0.0104	9.92	22.19
M2	0.0659	0.0106	0.0047	6.22	13.92
M4	0.0678	0.0106	0.0047	6.41	14.33

The M1, M2, and M4 effects are 9.92, 6.22, and 6.41 times the paired inference SD (22.19, 13.92, and 14.33 times the SE). Under this paired design, their differences are discernible beyond infer-

ence noise. In 20-seed repeats of full SCOPE, aggregate metrics have zero SD for Chronos-2 and TimesFM-2.5; Kairos's aggregate MAE SD is approximately 0.0000000352%.

Dataset variability addresses a different question. Across the four models, matched M1, M2, and M4 mean MAE gains are 0.2365%, 0.0191%, and 0.0198%. Dataset-level tests are nonsignificant after Holm correction, and Chronos-2 has slightly negative mean effects for M2 and M4. Thus the matched analysis supports small effects distinguishable from inference randomness, while the strongest evidence for projection remains improved average structural fidelity and conditional average forecast non-inferiority.

F Full Results for Model-aware Alignment

All SCOPE-ma results include selective activation and fallback. They should not be interpreted as the performance of unfiltered M5 updates.

F.1 Default SCOPE across Five Backbones

Table 29: Average ranks of five methods across TSFMs.

TSFM	Zero	Mean	Forward	Linear	SCOPE-mu	Best
Chronos-2	4.64	3.52	3.00	2.30	**1.54**	SCOPE-mu
Kairos-23M	4.47	3.47	3.16	2.14	**1.76**	SCOPE-mu
Sundial	4.53	3.65	3.04	2.25	**1.52**	SCOPE-mu
TimesFM-2.5	4.54	3.56	3.19	2.19	**1.52**	SCOPE-mu
VisionTS++	4.00	**2.12**	2.98	2.62	3.28	Mean

Before M5 activation, Table 29 gives SCOPE-mu average ranks of 1.54, 1.76, 1.52, and 1.52 on Chronos-2, Kairos-23M, Sundial, and TimesFM-2.5, respectively, ranking first on each. On VisionTS++ it falls to 3.28, behind Mean at 2.12 and Linear at 2.62. VisionTS++ consequently shows the clearest mismatch between model-unaware structural repair and downstream input preferences, making it a key test of M5.

F.2 VisionTS++: Ranks and Configuration-wise Outcomes

Table 30: Average MAE ranks in the VisionTS++ comparison.

Method	Overall	10% missing	20% missing	30% missing
Mean	**2.60**	**2.68**	2.53	2.59
SCOPE-ma	**2.60**	2.99	**2.37**	**2.46**
Linear	3.25	3.36	3.15	3.25
Forward	3.71	3.38	3.86	3.89
SCOPE-mu	4.03	4.10	4.19	3.80
Zero	4.80	4.48	4.90	5.01

Table 30 and Figure 10 show that SCOPE-ma improves the overall MAE rank from SCOPE-mu's 4.03 to 2.60, tying Mean for first. It ranks first at 20% and 30% missingness, with ranks of 2.37 and 2.46. At 10%, its rank of 2.99 remains worse than Mean's 2.68. Against SCOPE-mu, it records 183 wins and 60 losses, a 75.3% win rate, and mean and median MAE reductions of 4.79% and 7.76%.

Table 31: Head-to-head results for SCOPE-ma against five comparators on VisionTS++ (negative Δ favors SCOPE-ma).

Comparison	Wins	Losses	Win rate	Mean Δ MAE	Median Δ MAE
SCOPE-ma vs Mean	117	126	48.1%	-0.78%	+0.02%
SCOPE-ma vs Linear	153	90	63.0%	-0.98%	-1.51%
SCOPE-ma vs Forward	177	66	72.8%	-2.79%	-5.96%
SCOPE-ma vs Zero	195	48	80.2%	-27.27%	-18.63%
SCOPE-ma vs SCOPE-mu	183	60	75.3%	-4.79%	-7.76%

Against Linear, Forward, and Zero, SCOPE-ma wins 63.0%, 72.8%, and 80.2% of configurations, respectively. Against Mean, its win rate is only 48.1%, although average MAE decreases by 0.78% and the median change is $+0.02\%$. A smaller number of larger improvements therefore offsets slightly more numerous but smaller regressions. M5 substantially mitigates the VisionTS++ input mismatch and brings SCOPE-ma level with Mean in aggregate rank; gains over Mean are concentrated in the larger improvements rather than a majority of configurations.

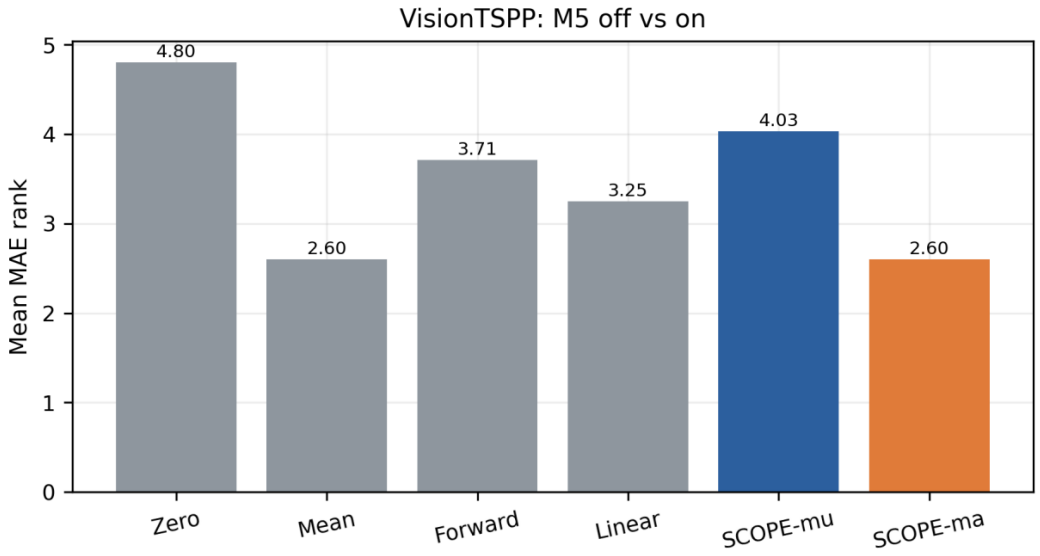

Figure 10: VisionTS++ performance before and after M5 activation.

F.3 Cross-backbone Fallback and Structural Sensitivity

On Chronos-2, Kairos-23M, Sundial, and TimesFM-2.5, all 243 configurations exactly match SCOPE-mu, giving zero mean and median MAE changes. VisionTS++ yields 183 wins, 60 losses, and a 4.79% average reduction. The final strategy therefore improves VisionTS++ while preserving the other four backbones through fallback. These results evaluate the gated strategy. Transfer to a new TSFM relies on independent calibration to choose activation or fallback.

Figure 12 shows TS-dominated weights for Chronos-2, Sundial, and TimesFM-2.5: 0.704, 0.745, and 0.727. Kairos-23M emphasizes SS at 0.554, followed by TS at 0.255. VisionTS++ instead emphasizes SE at 0.619 and RA at 0.187. These profiles are consistent with the VisionTS++ gains and fallback on other models, but the weights themselves explain preferences rather than independently prove general M5 effectiveness.

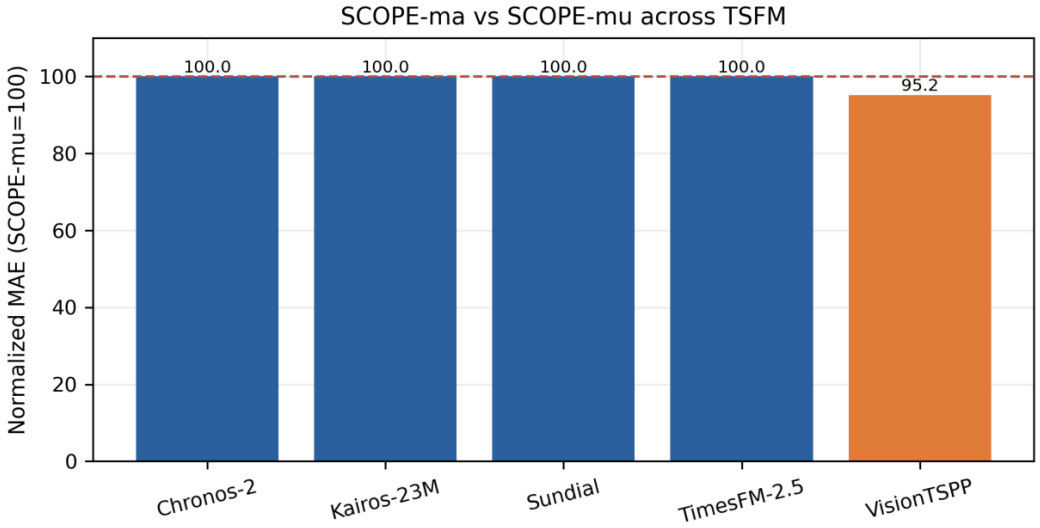

Figure 11: SCOPE-ma MAE normalized to SCOPE-mu across five TSFMs.

Table 32: Head-to-head results for SCOPE-ma against SCOPE-mu across five TSFMs.

TSFM	Wins	Losses	Win rate	Mean Δ MAE	Median Δ MAE
Chronos-2	0	0	— (243 ties)	+0.00%	+0.00%
Kairos-23M	0	0	— (243 ties)	+0.00%	+0.00%
Sundial	0	0	— (243 ties)	+0.00%	+0.00%
TimesFM-2.5	0	0	— (243 ties)	+0.00%	+0.00%
VisionTS++	183	60	75.3%	-4.79%	-7.76%

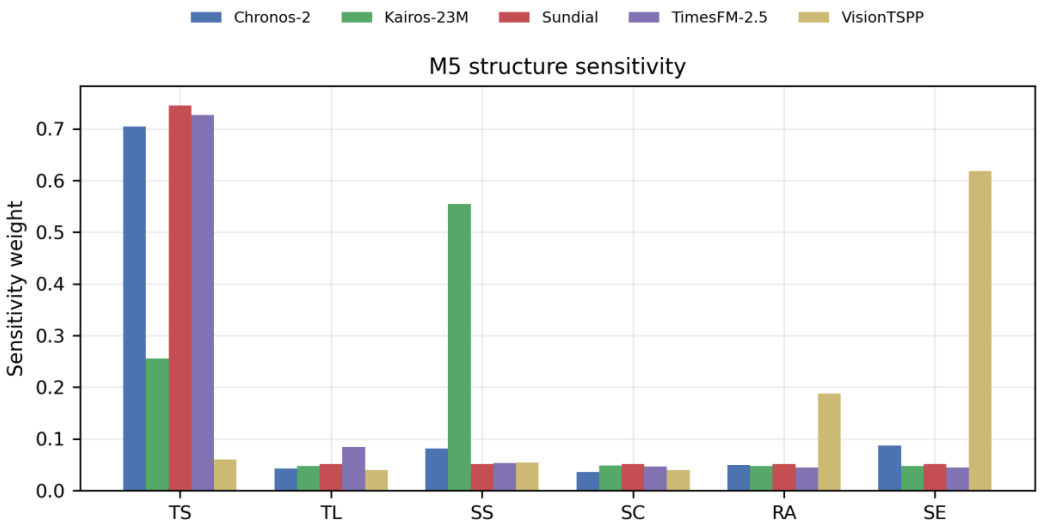

Figure 12: M5 structural-sensitivity weights for each TSFM.