CAN TIME-SERIES FOUNDATION MODELS SERVE AS SIMULATORS FOR PROCESS CONTROL?

Anonymous authors
Paper under double-blind review

ABSTRACT

Can a time-series foundation model (TSFM) predict what happens when we change an input, and help choose actions that keep an output near its target? We introduce a benchmark covering five synthetic process families and twelve evaluation factors for prescribed-input prediction, with executed control tests on selected conditions from all five families. For each proposed action, known dynamics generate two trajectories from the same state: one with the action and one with the input held unchanged. Their difference reveals the action response. We assess that response, then use each predictor to select inputs in an executed control test. We compare three frozen TSFMs with four locally fitted models. On clean first-order systems with input delays, TSFM predictions include transient reversed responses, timing errors, and amplitude errors. On integrating processes, their response-skill point estimates exceed the tested finite-impulse-response and multilayer-perceptron models, and all three also achieve lower tracking error. A locally fitted autoregressive model nevertheless has the lowest family-equal core tracking error among learned models. Better prediction does not guarantee better control: among local models with identical fitted artifacts reused across stages, orderings reverse in 10.2% of weighted within-task comparisons on clean first-order tasks, retaining ties in the denominator. The benchmark evaluates physical response fidelity together with executed control, exposing capabilities and failures that forecasting scores alone do not capture. Benchmark resources: ProSimCtrlBench (anonymous).

1 INTRODUCTION

A simulator for control must predict what happens when we change an input. Consider a tank whose liquid level should stay near a target. The tank is the *process*; a *predictor* estimates future levels for proposed inflows; a *controller* uses those predictions to choose an inflow sequence. Time-series foundation models (TSFMs) that accept future-known covariates can receive proposed inputs directly (Ansari et al., 2025; Google Research, 2026; Podest et al., 2026), offering a way to simulate processes from supplied histories without training a separate model for each process.

Yet continuing an observed trajectory and predicting the effect of a changed input are different tasks. In a feedback-controlled process, inputs reflect the controller's reactions to earlier outputs and targets (Åström & Hägglund, 2006; Ljung, 1999). A model may reproduce this familiar behavior while getting the response to a new action wrong. Even a small trajectory error can conceal an early response, a slow rise, or an incorrect amplitude. These errors matter because the controller acts on the predicted consequences of its choices.

Existing benchmarks establish broad forecasting coverage, as in GIFT-Eval, and evaluate covariate-aware forecasting, as in fev-bench (Aksu et al., 2024; Shchur et al., 2025). Our focus is the additional step from a forecast to an action: predicting alternative responses from the same physical state, then testing the decisions made from those predictions.

We ask: **Do TSFM predictions preserve physical response properties? How effectively do they support closed-loop control? Do better predictions translate into better control?**

We apply proposed inputs to known dynamics from the same initial state, then use each predictor inside a shared model-predictive control (MPC) procedure. The benchmark spans five process families and twelve factors varying dynamics, input histories and plans, observation quality, and

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

loads. Three frozen TSFMs use their native covariate interfaces without task-specific weight updates; four baselines are fitted only to the supplied history.

On clean first-order systems, TSFM predictions include transient reversed responses as well as timing and amplitude errors. Long delays and aggressive feedback histories reduce accuracy; integration is a relative strength compared with the tested finite-impulse-response and multilayer-perceptron models. A locally fitted autoregressive model leads family-equal core tracking, while nonlinear outflow exposes physical terminations. Prediction and control orderings can reverse even among local models with verified fitted-artifact reuse.

Our contributions are threefold. First, the benchmark provides a process generator, paired action/hold references, and grouped evaluation conditions across five families. Second, a shared protocol measures physical response fidelity and executed control using the same predictor interface. Third, a paired analysis tests where prediction and control orderings disagree. The public software includes a reference workflow for adding predictors and executing control; reproducing the paper protocol additionally requires its frozen data and model bindings (Appendix G).

2 Evaluating a simulator for control

We evaluate prediction and execution separately (Figure 1). Let y_t denote process output (PV), u_t the manipulated input (MV), and v_t an external load (DV). Given a 1,024-step history $\mathcal{H}_t$ and a proposed input sequence, a predictor produces

$$\hat{y}_{t+1:t+H} = f(\mathcal{H}_t, u^{\text{plan}}_{t+1:t+H}, v^{\text{visible}}_{t+1:t+H}). \tag{1}$$

Here, H is the forecast horizon; we use $H = 96$ steps throughout the benchmark. The load channel is available only in measured-DV conditions. A proposed input is a choice the controller can make; a future load is not. During control, the first load event is unannounced. Once a load is observed, its most recent value is held constant over the forecast horizon, rather than revealing its future trajectory. Setpoints (target outputs) define the controller's objective but are not inputs to the process predictor. Neither future outputs nor hidden physical states are supplied to learned models.

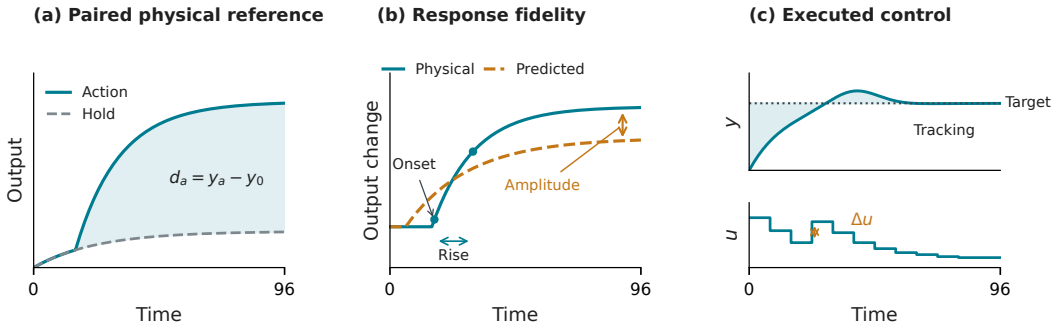

Figure 1: **From predicted responses to executed control.** (a) Apply an action and a hold input from the same state; their output difference isolates the incremental action response. (b) Assess response direction over time, together with onset, rise interval, and tail amplitude. (c) Execute model-selected inputs and measure target-tracking error and input variation. Curves are schematic.

Does the model predict the response to an action? For each proposed action a, we simulate both its output y_a and the output y_0 obtained by holding the last input. Their difference $d_a = y_a - y_0$ is the action's incremental response; the model predicts $\hat{d}_a = \hat{y}_a - \hat{y}_0$. This subtraction matters: holding the input does not necessarily hold the output constant. Using RMS error over the 96-step horizon, we define

$$E_{PV} = \text{RMS}(\hat{y}_a - y_a), \qquad B_{PV} = \text{RMS}(b - y_a), \tag{2}$$

$$E_{MV} = \text{RMS}(\hat{d}_a - d_a), \qquad B_{MV} = \text{RMS}(d_a), \tag{3}$$

$$\text{PV skill} = 1 - \overline{E}_{PV}/\overline{B}_{PV}, \qquad \text{MV skill} = 1 - \overline{E}_{MV}/\overline{B}_{MV}, \tag{4}$$

where b is the last observed output. PV skill measures output prediction against persistence; action-response skill (MV skill) measures incremental-output prediction against a zero-response reference.

For both, one is perfect, zero matches the reference, and negative values indicate worse performance. We average errors and reference magnitudes before forming each ratio. Reporting both scores avoids rewarding a model solely because a shared bias cancels between action and hold predictions.

We diagnose response direction, onset, rise interval, and amplitude on clean first-order-plus-dead-time (FOPDT) tasks. For direction, we report the fraction of time steps with a reversed incremental response, starting when the true physical response begins. Appendix F separately reports premature and weak responses, so zero response is not treated as correct. Tail amplitude is the mean of the final 24 steps; onset and rise interval use the first crossings of 5% and 63.2% of each response's own tail amplitude. These are finite-window features, not estimates of asymptotic physical parameters.

Do the predictions support effective control? MPC searches for an input plan, executes its first four steps, and replans from the resulting observations. The model is not retrained during execution. For target r_t, we report normalized tracking error and input variation,

$$\mathrm{nIAE} = \frac{1}{T s_y} \sum_{t=1}^{T} |y_t - r_t|, \quad \mathrm{nTV} = \frac{1}{T s_u} \sum_{t=1}^{T} |u_t - u_{t-1}|, \quad J = \mathrm{nIAE} + \lambda_{\mathrm{TV}}\, \mathrm{nTV}, \tag{5}$$

where T is the control evaluation length and λ_{TV} weights input variation relative to tracking error. We use $T = 96$ steps and $\lambda_{\mathrm{TV}} = 0.05$. The scales s_y and s_u are the historical 95th–5th percentile ranges. Lower values are better. A physical boundary violation terminates execution. We compare error on tasks completed by every configuration and report terminations separately; completion-versus-error curves additionally retain all tasks, including terminated runs.

3 ProSimCtrlBench: Data and Software

ProSimCtrlBench spans **five process families and twelve evaluation factors**, with multiple settings within each factor (Figure 2). Each query pairs a history and input plan with a known physical response. The benchmark targets predictor developers testing action response and fixed-controller performance across controlled SISO mechanisms. Appendix B maps factors to families and executed tests.

Five response families. First-order-plus-dead-time (FOPDT) processes respond after a delay and gradually settle. In a mixing tank, feed flow changes concentration; in an outflow tank, inflow changes liquid level while outflow depends on that level. Both nonlinear families vary response size and speed with operating level. Second-order-plus-dead-time (SOPDT) processes allow overshoot and oscillation. Integrating processes accumulate sustained input imbalance, so their outputs keep changing rather than settling. All five families have one manipulated input and one measured output.

Twelve factors with multiple settings. Seven factors vary the *process response*: time constant (response speed), dead time (delay), gain (output change per input change), mixing and outflow nonlinearity, second-order damping, and integration. Each uses three parameter levels or ranges, not a common difficulty scale across models.

Three factors vary *historical and future inputs*. Controller-history tests include an open-loop reference and conservative, nominal, and aggressive proportional–integral (PI) histories; PI adjusts inputs using current and accumulated target error. Numerical input extrapolation compares future inputs inside and outside the historical range. History–query pattern tests use step, ramp, pulse, and sinusoidal inputs.

Two factors vary *observations and loads*: five corruption types affect three fractions of historical PV; load tests use three strengths and a clean reference, with measured and hidden DV views. Figure 2c lists the settings.

Paired construction and dataset scale. Action and hold branches start from the same complete physical state, including delayed inputs, so their output difference measures the effect of the action. Observation corruption changes historical PV while retaining the future physical truth. Loads instead change the process equations; measured and hidden views expose different information about the same load. Prediction history-pattern tests approximately match terminal states and use identical

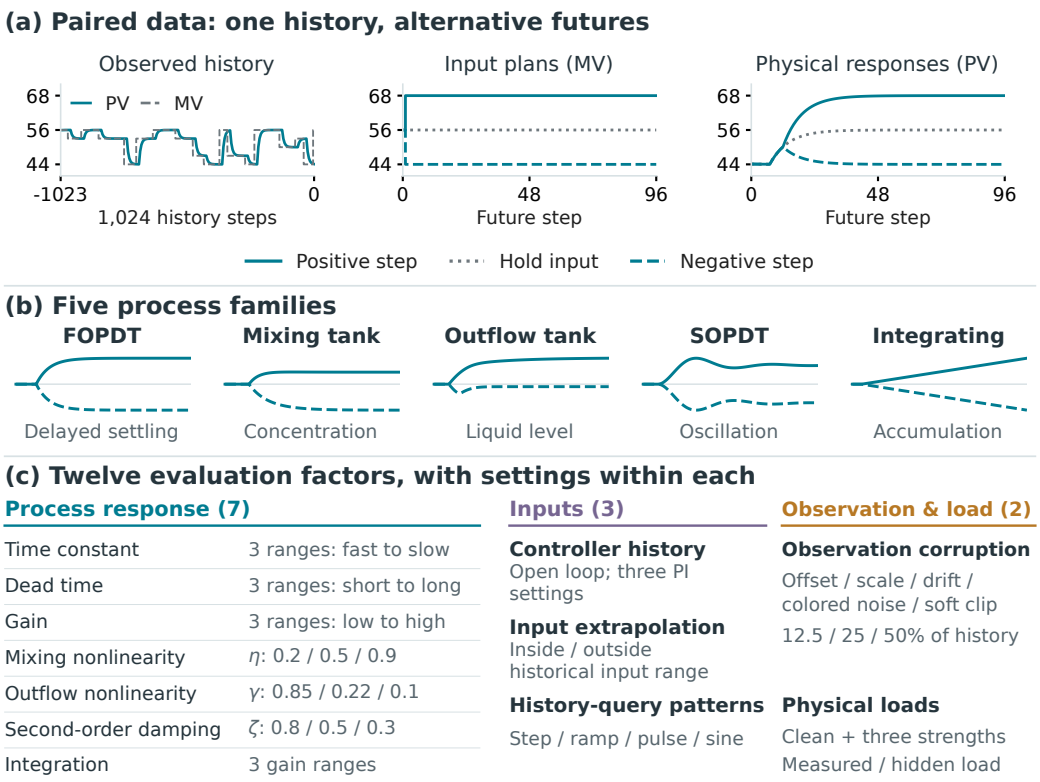

Figure 2: **Paired data and evaluation coverage.** (a) A recorded FOPDT history, two of its six action plans, and the hold reference; physical outputs provide evaluation truth. (b) Positive (solid) and negative (dashed) action-minus-hold responses over 96 steps for one fixed instance per family, scaled within each pair; these are examples, not population averages. (c) Twelve factors with internal settings. Coverage is targeted, not a full Cartesian product; Appendix B gives exact ranges and family coverage.

future inputs. Control history-pattern tests retain their observed states, so they do not isolate the effect of history alone.

A sampling group identifies the source of related conditions. Its parameterized process instances supply histories, each with multiple future-input queries. The simulation inventory contains 1,000 sampling groups, 2,200 process instances, 17,100 history conditions, and 135,700 queries per model, including hold references. Derived observation and information conditions retain their source-group identity and do not count as independent processes, supporting grouped comparisons across conditions and models.

Reference validation and response coverage. Physical references are checked against analytical or independent numerical solutions. Slow and integrating responses remain eligible for full-horizon scoring even without a post-transient window. Appendix B gives parameter ranges, construction counts, validation checks, and response-window statistics.

Reusable software and interactive analysis. Researchers configure conditions, plans, seeds, and sample counts, then connect a predictor through a history-and-query adapter. The formal comparison uses the frozen task inventory, settings, and weights in Section 4 and Appendix C. The separate reference workflow supports smaller configurable tests and an explorer (Appendix G); its defaults do not reproduce the paper leaderboard.

The explorer filters conditions, compares responses, and replays recorded control. Exports retain scores, failures, and source identities; saved metrics support reaggregation. Appendix G gives commands and release coverage.

4 EXPERIMENTAL SETUP

Predictors. We compare frozen Chronos-2, TiRex-2, and TimesFM-3 with four history-fitted models. The autoregressive model with exogenous inputs (ARX) uses past outputs and inputs. The finite impulse response (FIR) model, multilayer perceptron (MLP), and radial-basis-function support vector regressor (RBF-SVR) use input-history features without PV feedback. TSFMs use native covariate adapters without weight updates (Appendix C.2). Local models select configurations by rolling-origin historical validation and refit on that history, without future query truth or control-test outcomes. These are different adaptation regimes, not equal training budgets; findings concern the tested implementations.

Prediction tests. All seven models complete 135,700 queries: 118,600 actions and 17,100 hold references. A prediction query specifies one future input plan; a control task fixes a starting history, process, information view, and target, and can have several prediction probes. An evaluation cell groups queries by declared test conditions for scoring; it is not an independent process. Table 1 maps the main analyses to their populations. Overall prediction scores are point estimates; parameter contrasts use paired group bootstrap intervals.

Table 1: **Populations and aggregation.** Counts are per model or configuration. Physical core is a prediction subset; core control is part of 5,594 test tasks. Paired FOPDT joins core and nominal controls to in-range prediction probes and supplies the response-feature diagnostics. Rows overlap and use comparison-specific weights.

Analysis	Support	Aggregation
Overall prediction	118,600 non-hold actions	Evaluation cells equally
Physical core	26,400 step responses from 4,400 histories	Within, then across groups
Core control	2,228 tasks; 2,166 common complete	Groups within strata; five families equally
Paired clean FOPDT	940 tasks, 376 groups, 2,994 probes	Pairs, levels, groups, and sources equally at each layer

Control tests. We use the seven predictors inside a shared MPC procedure and add two references: true-model MPC and an ARX-tuned PI controller. MPC uses a 96-step horizon, eight action blocks, and cross-entropy search with 32 candidates in each of two rounds. It replans every four steps. True-model MPC uses the same search budget with privileged state information; it is a finite-budget reference, not an optimal-control bound. PI is tuned on historical ARX-surrogate rollouts and receives feedback every step, so its comparison also reflects a different feedback frequency.

Excluding 70 development or previously inspected groups leaves 930 groups and 5,594 control tasks per configuration, covering all five families, four history patterns, historical soft clipping, and physical loads. Core error means use common completion across all nine configurations; termination measures retain all tasks. The 95% intervals resample matched groups together. Six learned-model comparisons against ARX use Holm correction; other contrasts and the paired prediction–control analysis are exploratory. Appendix C gives the inventory, execution rules, weights, and failure treatment.

5 RESULTS

5.1 DO TSFM PREDICTIONS PRESERVE PHYSICAL RESPONSE PROPERTIES?

Different dynamics favor different models. ARX, FIR, and MLP lead the overall simulation scores (Table 2). That ranking, however, hides important differences between families. ARX and FIR are nearly exact on stable linear systems, which match their model structure. MLP predicts the nonlinear tank responses more accurately than ARX. Integration reverses part of this pattern: all three TSFMs have higher aggregate response-skill point estimates than the tested FIR and MLP models, with TimesFM-3 reaching MV skill 0.636 versus 0.170 for FIR and -0.712 for MLP (Figure 3a). Chronos-2 is only slightly above FIR (0.18 versus 0.17); no paired significance claim is made for this difference. ARX remains near one. The input-history representations used by FIR and MLP can limit their treatment of accumulation; these results do not characterize every possible implementation of those model classes.

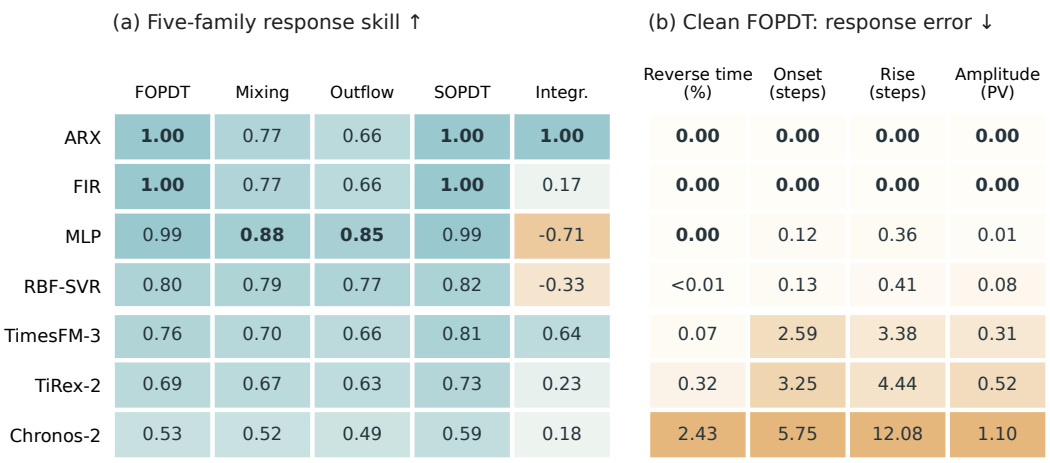

Figure 3: **Response accuracy depends on the physical mechanism.** (a) MV skill on the five-family physical core; higher is better. (b) Diagnostics on paired clean-FOPDT probes; lower is better. Reverse time is the weighted percentage of steps with a reversed incremental response after the physical delay. Shading in (b) is scaled separately by column. Values are descriptive means; bold marks the displayed best values, not significance. The panels use different populations.

Response errors include transient reversals. On clean FOPDT, 121 of 2,994 TimesFM-3 probes contain a reversed response after the physical delay; 48 exceed 5% of the true step amplitude. Reversed-response duration averages 0.07%, 0.32%, and 2.43% for TimesFM-3, TiRex-2, and Chronos-2, respectively (Figure 3b). These values average time fractions within probes, rather than counting affected probes. Timing and amplitude errors also remain: TimesFM-3's mean onset and rise-interval errors are 2.59 and 3.38 steps, versus 5.75 and 12.08 for Chronos-2; their tail-amplitude errors are 0.31 and 1.10 PV units.

Table 2: **Overall simulation accuracy.** Scores aggregate 118,600 non-hold actions per model with equal evaluation-cell weights. E is mean RMSE in PV units. Values are point estimates; no aggregate leaderboard intervals are available.

Model	PV skill ↑	MV skill ↑	E_{PV} ↓	E_{MV} ↓
ARX	0.8868	0.8856	0.6086	0.5735
FIR	0.8531	0.8716	0.7896	0.6435
MLP	0.8322	0.8353	0.9023	0.8258
RBF-SVR	0.7235	0.7189	1.4868	1.4092
TimesFM-3	0.7396	0.7236	1.4000	1.3859
TiRex-2	0.6954	0.6927	1.6379	1.5405
Chronos-2	0.5428	0.5362	2.4586	2.3254

Timing and historical excitation are harder than rescaling. Longer delays sharply reduce TSFM response accuracy (Figure 4). TimesFM-3's MV skill falls from 0.898 to 0.620, TiRex-2's from 0.698 to 0.422, and Chronos-2's from 0.610 to 0.117. All three long-minus-short group intervals exclude zero. This is not merely a normalization effect: raw response RMSE also increases. By contrast, changing gain alters each TSFM's skill by less than 0.003 across the tested levels.

Aggressive PI histories also reduce accuracy relative to nominal histories, by 0.406 for TimesFM-3, 0.506 for TiRex-2, and 0.308 for Chronos-2. Such histories change input coverage, oscillation, and terminal state together, so this comparison identifies a difficult history regime rather than a single cause. The complete twelve-panel results and paired sensitivity intervals appear in Appendix C.

5.2 How effectively do TSFMs support closed-loop control?

ARX leads family-equal core tracking among learned models. Under shared MPC, ARX achieves nIAE 0.235, compared with 0.345 for TiRex-2, 0.366 for TimesFM-3, and 0.562 for Chronos-2 (Figure 5). All six learned-model comparisons against ARX remain significant after Holm correction. TiRex-2 leads the TSFMs on this aggregate, although its difference from FIR is less clear:

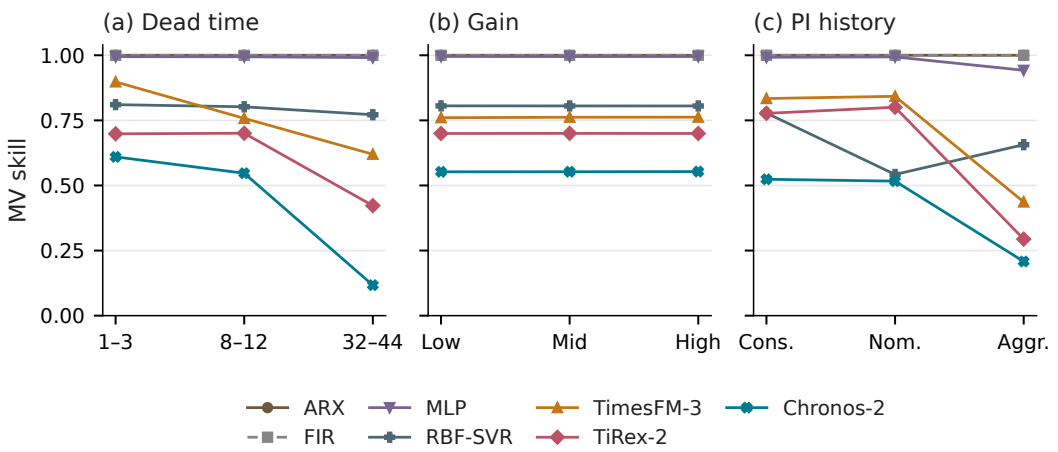

Figure 4: **Delay and aggressive feedback histories reduce TSFM response accuracy.** Group-equal MV skill across three sampled levels of each factor, with 100 matched groups per axis. Gain changes have little effect on normalized accuracy. Lines connect sampled levels; ARX and FIR overlap near one.

$\Delta\text{nIAE} = 0.0105$, with a 95% interval of $[-0.0017, 0.0226]$. TSFMs also use more input variation: nTV 0.573–1.132 versus ARX's 0.225. Including this penalty gives $J = 0.374$–0.619 versus 0.246 (Appendix E).

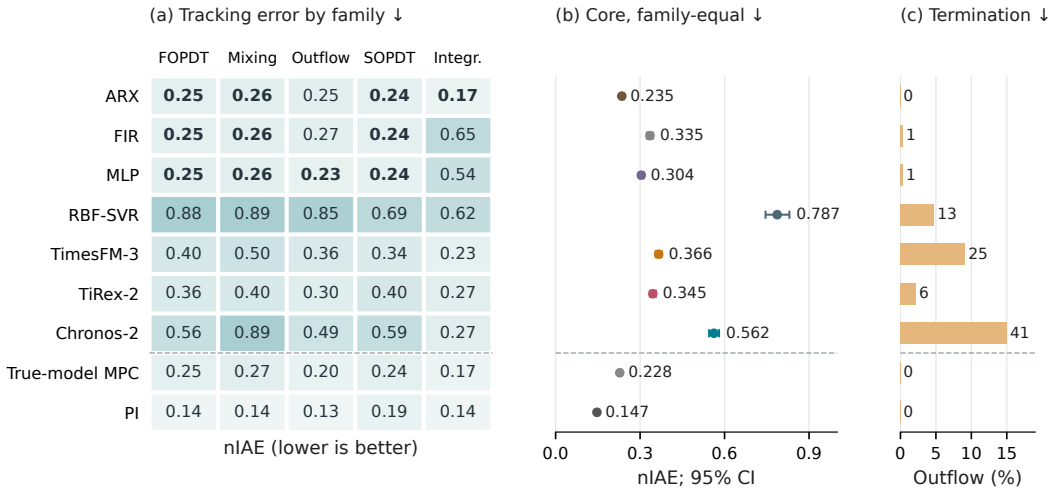

Figure 5: **Executed control varies across families, and low mean error can hide termination.** (a) Family-specific nIAE on tasks completed by all configurations. A common color scale is used; bold marks the best displayed learned-model mean per family. (b) Family-equal core mean and 95% paired-group interval, based on 2,166 common complete tasks; labels give means. (c) Bars show raw termination percentages among all 273 outflow tasks; labels give counts. References below the line use the distinct information and feedback protocols in Section 4.

The advantage on integration survives execution. All three TSFMs also outperform the tested FIR and MLP implementations on integrating processes: nIAE 0.235–0.270 versus 0.651 and 0.542. ARX remains the best learned model. This agreement with prediction diagnoses performance under the tested state representations; it does not isolate an advantage due to pretraining.

Outflow tanks expose a different failure mode. On completed outflow tasks, MLP has lower mean error than ARX (0.227 versus 0.253). Yet only 212 of 273 outflow tasks complete under every configuration. Chronos-2 terminates on 41 tasks, TimesFM-3 on 25, and TiRex-2 on six. Across all test modules, 112 of 113 configuration-level terminations occur in outflow tanks. These failures are

part of the control result, so Figure 5 reports them alongside the conditional error means. Appendix E provides completion curves that retain every task.

Model fidelity is only one part of the control procedure. PI achieves nIAE 0.147, below even the true-model MPC's 0.228. PI receives feedback every step, while MPC replans every four steps and uses finite search. The result therefore reflects the planning and feedback protocol as well as the predictor; it does not isolate the contribution of feedback frequency or search budget.

Observing a load does not help every predictor. A separate stress test varies whether the external load is visible. Measured DV lowers event-period tracking error for ARX, FIR, MLP, and Chronos-2, but raises it for TimesFM-3 and RBF-SVR. Access to a relevant signal therefore does not guarantee that a model uses it effectively under the tested interface. These paired information-view contrasts, together with observation-corruption results, are reported in Appendix E.

5.3 Do better predictions translate into better control?

We compare prediction and control on 940 clean FOPDT tasks, matching history, process instance, and information view. Prediction uses in-range step probes; MPC can visit different inputs and states. All configurations complete these tasks. The comparison therefore tests ranking transfer under this fixed protocol, without isolating the effects of distribution change or finite search.

Reversals persist with identical fitted predictors. Among the four local models, whose fitted artifacts are verified and reused in control, output-RMSE and nIAE orderings reverse in 10.21% of weighted comparisons. This directly supports a prediction–control mismatch under the tested protocol; a subset confidence interval has not been computed.

Historical seven-model comparisons show a larger reversal share. Across 19,740 pairs of recorded configurations, 62.4% align, 23.2% reverse, and the rest contain a numerical tie (Appendix F). The exploratory group interval is $[22.2, 24.2]\%$; excluding ties gives 27.1%. TSFM historical runtime identity across stages is not fully verified (Appendix C.2), so this broader comparison has weaker identity evidence than the local subset. MAE and incremental-response errors give similar rates. Excluding RBF-SVR leaves 16.34% reversals.

Within reversed pairs, mean excess nIAE is 0.395 ($[0.362, 0.429]$), with weighted median 0.129. RBF-SVR–TSFM pairs contribute 83.06% of its weighted numerator, so the mean is not typical of every model pair. This is a conditional difference, not best-model selection regret. Figure 6 illustrates one recorded reversal; Appendix F gives selection rules and full decompositions.

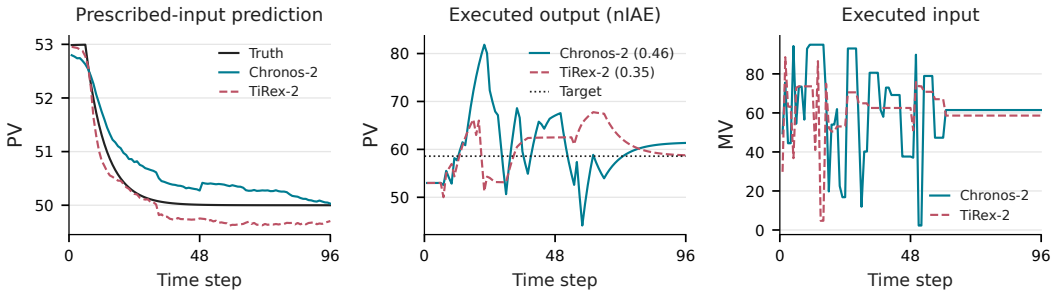

Figure 6: **A recorded prediction–control reversal.** Nominal FOPDT case closest to the source-weighted median reversal penalty. The solid configuration has lower mean probe error but higher executed nIAE (legend). The left panel shows one probe; control chooses different inputs. This historical-configurations example illustrates behavior, not its prevalence or cause.

6 Related work

Forecasting benchmarks. GIFT-Eval evaluates forecasts across domains (Aksu et al., 2024), TIME examines temporal structure alongside scalar error (Qiao et al., 2026), and BOOM studies production telemetry (Cohen et al., 2025). The fev-bench benchmark includes covariate-aware tasks (Shchur

et al., 2025), while MUSEval compares univariate and multivariate forecasting (Synthefy Research Team, 2025). Our benchmark instead evaluates alternative input plans from the same physical state, measuring both output and incremental-response accuracy before executing model-selected actions through shared MPC.

Covariate interfaces and response identification. Chronos-2, TimesFM-3, and TiRex-2 accept future-known covariates (Ansari et al., 2025; Google Research, 2026; Podest et al., 2026), allowing us to supply proposed inputs without a task-specific learned adapter. This permits prescribed-input prediction without establishing response identification from feedback-generated histories, where inputs depend on earlier outputs (Ljung, 1999). We evaluate paired action/hold responses and vary historical inputs using approximately terminal-state-matched simulation. These comparisons concern known dynamics, not causal identification from arbitrary observational data.

System identification and control benchmarks. Resources such as Silverbox provide measured input–output records for nonlinear identification (Wigren & Schoukens, 2013). BOPTEST supplies building emulators and common control metrics (Blum et al., 2021); studies using it already compare surrogate prediction and MPC performance (Mostafavi et al., 2023). Our contribution is a controlled SISO mechanism suite with same-state action/hold references, native covariate TSFM interfaces, and grouped prediction–execution comparisons. Its scope complements measured-system identification and building-level controller evaluation.

Prediction models used for decisions. Building-energy evaluations examine TSFM generalization and covariate use (Mulayim et al., 2025), while learned-dynamics studies assess whether fitted models reproduce long-term statistical properties (Park et al., 2024). Beyond these prediction evaluations, causal forecasting trains models to predict action effects (Crasson et al., 2024); decision-focused identification and task-aware Koopman methods incorporate downstream objectives into model learning (Favaro et al., 2025; Mayfrank et al., 2024). ThermoForce combines a frozen TSFM prior with a learned physics-structured response operator for heating, ventilation, and air-conditioning effects and MPC (Wang, 2026). We evaluate native covariate TSFMs without a separately learned response operator alongside history-fitted baselines, comparing response fidelity and executed control across physical mechanisms and their within-task orderings on clean FOPDT.

7 Discussion and conclusion

Frozen TSFMs can predict responses to proposed inputs and support MPC, with capabilities that depend on the dynamics. They handle integration better than some tested local representations. On clean first-order systems, their predictions include transient reversed responses, timing errors, and amplitude errors. These coexist with higher family-equal core tracking error than ARX and physical terminations on nonlinear outflow tasks.

These conclusions concern synthetic dynamics, the tested implementations and adaptation regimes, and one search and feedback protocol; ranking stability under other budgets remains untested. Control covers selected conditions, with error means conditional on completion. Clean-FOPDT reversals among bound local predictors support the prediction–control distinction; the broader seven-model analysis compares historical configurations with incomplete TSFM runtime identity. Transfer to real processes remains open.

AI use statement

Generative AI was used for manuscript writing, language editing, and revision-stage code and analysis checks.

Reproducibility statement

Sections 2–4 specify the data, predictor inputs, scores, and control protocol. Appendices B–F give construction details, populations, model settings, aggregation, and failure treatment. The accompanying source distribution for `https://anonymous.4open.science/r/ProSimCtrlBench-34D0` contains the reference workflow, formal research runners, original

generation parameters, inventory/split checks, and a formal execution guide. The manuscript sources contain aggregate statistics with source hashes and scripts regenerating figures and tables, including the model-pair reversal analysis. Appendix G reports verified inventory reconstruction and a five-family ARX check. Complete numerical archives are not bundled, and a new seven-model reproduction has not been verified end to end. Historical model-version gaps remain documented in Appendix C.2.

Ethics statement

The study uses synthetic process trajectories and foundation-model checkpoints. It does not involve human subjects, personal data, or deployment on a physical plant. The findings concern the tested configurations and do not certify closed-loop deployment safety.

References

Taha Aksu, Gerald Woo, Juncheng Liu, Xu Liu, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. GIFT-Eval: A benchmark for general time series forecasting model evaluation. *CoRR*, abs/2410.10393, 2024. doi: 10.48550/ARXIV.2410.10393. URL https://doi.org/10.48550/arXiv.2410.10393.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting. *CoRR*, abs/2510.15821, 2025. doi: 10.48550/ARXIV.2510.15821. URL https://doi.org/10.48550/arXiv.2510.15821.

Karl Johan Åström and Tore Hägglund. *Advanced PID Control*. ISA, Research Triangle Park, NC, 2006.

David Blum, Javier Arroyo, Sen Huang, Ján Drgoňa, Filip Jorissen, Harald Taxt Walnum, Yan Chen, Kyle Benne, Draguna Vrabie, Michael Wetter, and Lieve Helsen. Building optimization testing framework (BOPTEST) for simulation-based benchmarking of control strategies in buildings. *Journal of Building Performance Simulation*, 14(5):586–610, 2021. doi: 10.1080/19401493.2021.1986574. URL https://doi.org/10.1080/19401493.2021.1986574.

Ben Cohen, Emaad Khwaja, Youssef Doubli, Salahidine Lemaachi, Chris Lettieri, Charles Masson, Hugo Miccinilli, Elise Ramé, Qiqi Ren, Afshin Rostamizadeh, Jean Ogier du Terrail, Anna-Monica Toon, Kan Wang, Stephan Xie, Zongzhe Xu, Viktoriya Zhukova, David Asker, Ameet Talwalkar, and Othmane Abou-Amal. This time is different: An observability perspective on time series foundation models. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/48ea942b60e6c52b50ca90e4a40d8ac2-Abstract-Conference.html.

Thomas Crasson, Yacine Nabet, and Mathias Lécuyer. Training and evaluating causal forecasting models for time-series. *CoRR*, abs/2411.00126, 2024. doi: 10.48550/ARXIV.2411.00126. URL https://doi.org/10.48550/arXiv.2411.00126.

Pietro Favaro, Jean-François Toubeau, François Vallée, and Yury Dvorkin. Decision-focused learning for complex system identification: HVAC management system application. In *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems, E-Energy 2025, Rotterdam, The Netherlands, June 17-20, 2025*, pp. 347–358. ACM, 2025. doi: 10.1145/3679240.3734584. URL https://doi.org/10.1145/3679240.3734584.

Google Research. TimesFM 3.0: Official model card and documentation. https://huggingface.co/google/timesfm-3.0-pytorch, 2026. Interface documentation: https://github.com/google-research/timesfm. Accessed September 26, 2026.

Lennart Ljung. *System Identification: Theory for the User*. Prentice Hall, Upper Saddle River, NJ, 2 edition, 1999. ISBN 9780136566953. URL https://www.control.isy.liu.se/books/sysid/.

Daniel Mayfrank, Alexander Mitsos, and Manuel Dahmen. End-to-end reinforcement learning of Koopman models for economic nonlinear model predictive control. *Comput. Chem. Eng.*, 190: 108824, 2024. doi: 10.1016/J.COMPCHEMENG.2024.108824. URL https://doi.org/10.1016/j.compchemeng.2024.108824.

Saman Mostafavi, Chihyeon Song, Aayushman Sharma, Raman Goyal, and Alejandro E. Brito. Benchmarking model predictive control algorithms in building optimization testing framework (BOPTEST). In *Proceedings of Building Simulation 2023: 18th Conference of IBPSA*, volume 18. IBPSA, Sept 2023. doi: 10.26868/25222708.2023.1371. URL http://dx.doi.org/10.26868/25222708.2023.1371.

Ozan Baris Mulayim, Pengrui Quan, Liying Han, Xiaomin Ouyang, Dezhi Hong, Mario Bergés, and Mani Srivastava. Can time-series foundation models perform building energy management tasks? *CoRR*, abs/2506.11250, 2025. doi: 10.48550/ARXIV.2506.11250. URL https://doi.org/10.48550/arXiv.2506.11250.

Jeongjin Park, Nicole Yang, and Nisha Chandramoorthy. When are dynamical systems learned from time series data statistically accurate? In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/4dc57702c987e1e72f0dd2921edb5ded-Abstract-Conference.html.

Patrick Podest, Marco Pichler, Elias Bürger, Levente Zólyomi, Bernhard Voggenberger, Wilhelm Berghammer, Daniel Klotz, Sebastian Böck, Günter Klambauer, and Sepp Hochreiter. TiRex-2: Generalizing TiRex to multivariate data and streaming. *CoRR*, abs/2607.01204, 2026. doi: 10.48550/ARXIV.2607.01204. URL https://doi.org/10.48550/arXiv.2607.01204.

Zhongzheng Qiao, Sheng Pan, Anni Wang, Viktoriya Zhukova, Yong Liu, Xudong Jiang, Qingsong Wen, Mingsheng Long, Ming Jin, and Chenghao Liu. It's TIME: towards the next generation of time series forecasting benchmarks. *CoRR*, abs/2602.12147, 2026. doi: 10.48550/ARXIV.2602.12147. URL https://doi.org/10.48550/arXiv.2602.12147.

Oleksandr Shchur, Abdul Fatir Ansari, Caner Turkmen, Lorenzo Stella, Nick Erickson, Pablo Guerron, Michael Bohlke-Schneider, and Yuyang Wang. fev-bench: A realistic benchmark for time series forecasting. *CoRR*, abs/2509.26468, 2025. doi: 10.48550/ARXIV.2509.26468. URL https://doi.org/10.48550/arXiv.2509.26468.

Synthefy Research Team. MUSEval: Multivariate time series evaluation for foundation models. https://huggingface.co/datasets/Synthefy/MUSEval, 2025.

Yifan Wang. ThermoForce: A physics-structured interventional world model for building HVAC control. *CoRR*, abs/2607.03942, 2026. doi: 10.48550/ARXIV.2607.03942. URL https://doi.org/10.48550/arXiv.2607.03942.

Torbjörn Wigren and Johan Schoukens. Three free data sets for development and benchmarking in nonlinear system identification. In *12th European Control Conference, ECC 2013, Zurich, Switzerland, July 17-19, 2013*, pp. 2933–2938. IEEE, 2013. doi: 10.23919/ECC.2013.6669201. URL https://doi.org/10.23919/ECC.2013.6669201.

GUIDE TO THE SUPPLEMENTARY EVIDENCE

The supplement follows the questions raised by the main results. Appendix A explains how a forecast becomes a control decision. Appendix B describes the physical systems, and Appendix C specifies fitting and execution. Appendix D gives the complete physical-level and stress comparisons behind the simulation findings. Appendices E and F report executed outcomes and paired prediction–control evidence. Each comparison retains its own population.

A FROM A FORECAST TO A CONTROL DECISION

A.1 WHAT INFORMATION DOES THE PREDICTOR RECEIVE?

A process simulator predicts an output under a specified input. At a decision time, the controller has observed the history but can still choose the future input. We write the physical evolution as

$$x_{t+1} = F(x_t, u_t, v_t), \qquad y_t = g(x_t), \qquad u_t = \pi(\mathcal{H}_t, r_t), \tag{6}$$

where x_t includes all internal states and any pending input-delay queue. The learned predictor observes historical PV and MV, the proposed future MV plan, and DV only when its view permits it. It does not receive x_t, physical parameters, or future PV. The reference r_t determines which outputs the controller wants; it is not a process-predictor channel in the current evaluation.

Table 3: **Information roles in the current protocol.**

Quantity	Role
PV, y	Historical observations and future prediction target.
MV, u	Historical input and proposed future decision sequence.
DV, v	External load; supplied only in measured-DV views.
SV, r	Desired output, used by the controller to score a plan.
State, x	Private physical reference; available to true-model MPC only.
Hold branch	Physical continuation with the last MV maintained; it can have changing PV.

Feedback-generated history contains both the process response and the controller's reaction to it. By contrast, an open-loop query fixes its future inputs and does not adapt them to future observations. An open-loop query can therefore start after a closed-loop history. The distinction concerns how future inputs are chosen, not whether the preceding history contains feedback.

A.2 WHY IS THE HOLD BRANCH NEEDED?

Suppose the last observed PV is 50, the future hold response reaches 55, and the action response reaches 63. The action contributes $63 - 55 = 8$ at that time. Subtracting the historical endpoint gives 13, which includes motion already implied by the state and pending inputs. The physical action and hold trajectories must therefore share the complete initial state and load sequence. For a nonlinear process, their difference is a response to that particular intervention, not an assumption of linear superposition.

The same subtraction is applied to the model's two predictions. With $e_a = \hat{y}_a - y_a$ and $e_0 = \hat{y}_0 - y_0$,

$$\hat{d}_a - d_a = e_a - e_0. \tag{7}$$

A common offset can cancel even when both forecasts are inaccurate. Conversely, a model that ignores the new action can predict natural continuation reasonably well while having zero incremental-response skill. These examples explain why PV and MV skill are complementary, and why a true hold trajectory must not be substituted into the model's predicted response.

A.3 WHY DO RESPONSE TESTS AND CONTROL TESTS DIFFER?

For candidate set $\mathcal{U}_t$, the controller approximately solves

$$U_t^\star \in \arg\min_{U \in \mathcal{U}_t} J\big(\hat{Y}_t(U), U; R_t\big). \tag{8}$$

The cost J in Eq. (5) combines normalized tracking error with normalized input variation, weighted by λ_{TV}. Candidate predictions supply the output-dependent term; the executed trajectory supplies reported control error. The controller executes four actions before choosing again. Thus the final trajectory depends on repeated decisions, new observations, and the states reached by earlier actions.

Prescribed-input error measures how well the model predicts a fixed response, whereas closed-loop nIAE measures the trajectory produced by repeated selection and execution. A low open-loop error need not preserve small cost differences between competing plans. True-model MPC remains a finite-search reference rather than a global optimal-control bound.

A.4 How are scores aggregated and missing windows handled?

Let $E_{m,i}$ and B_i be a model error and its reference magnitude for action i. For declared cells $\mathcal{C}$, the main simulation score is

$$1 - \frac{|\mathcal{C}|^{-1} \sum_{c \in \mathcal{C}} |I_c|^{-1} \sum_{i \in I_c} E_{m,i}}{|\mathcal{C}|^{-1} \sum_{c \in \mathcal{C}} |I_c|^{-1} \sum_{i \in I_c} B_i}. \tag{9}$$

This is not the mean of individual action skills or the mean of the twelve displayed panel scores. Cells encode axis, family, level, information view, query pattern, actual amplitude or replacement identity, and direction. Repeated display of a reference condition does not create another independent sample. A zero reference magnitude makes skill undefined; the corresponding raw error remains meaningful.

All models are scored on the same 96-step full window. Stage diagnostics use the true delayed response and its settling time: pre-response, transient, and post-transient. Settling requires remaining within the specified band, not merely entering it once. A response that has not settled by step 96 retains its full-window score but lacks a post-transient score; integrating processes have no finite settling response. In the expanded simulation inventory, full, pre, transient, and post scoring cover 118,600, 83,400, 83,400, and 78,319 actions per model, respectively. These different supports prevent treating their means as a single typical trajectory. Ramp, pulse, and sinusoidal queries use full-window scoring without an imposed step-response decomposition.

B Expanded dataset specification

Table 4 enumerates the 12 construction components in the primary inventory. This is a construction inventory, not a second list of the twelve evaluation factors in Section 3: it lists the FOPDT anchor separately and combines the two nonlinear families in one component, whereas evaluation treats mixing and outflow nonlinearity separately. The common sampling interval is one, context length is 1,024, and forecast horizon is 96. These are targeted modules with declared reuse, not all combinations of the listed settings. The four-pattern module uses FOPDT, mixing, outflow, and SOPDT systems; physical loads use FOPDT, outflow, and SOPDT systems. Integrating processes form their own module.

Table 4: **Twelve construction components.** Parameter levels specify coverage within a component, not equal difficulty across components or models.

ID	Component	Settings and interpretation
1	FOPDT anchor	$K = 1$, $\tau \in [6, 10]$, integer $\theta \in [6, 14]$; interpretable lag and delay.
2	Time constant	$\tau \in [3, 4.5]/[6.5, 10]/[16, 20]$, at $K = 1, \theta = 10$.
3	Dead time	Integer $\theta \in [1, 3]/[8, 12]/[32, 44]$; background $\tau \sim U(6, 10)$ shared within each matched group.
4	Control history	Open-loop reference and conservative, nominal, aggressive PI histories; future queries remain prescribed input plans.
5	Gain	$K \in [0.4, 0.6]/[0.85, 1.2]/[1.7, 2.2]$, at $\tau = 8, \theta = 10$.
6	Nonlinear dynamics	Mixing $\eta = 0.2/0.5/0.9$; outflow $\gamma = 0.85/0.22/0.1$; matched nominal local gain and time scale.
7	Second-order damping	SOPDT $\zeta = 0.8/0.5/0.3$; complete internal state retained, including the second-order state.
8	Numerical input shift	Inside/outside labels from actual future MV relative to the context range; 1,936 available numerical contrast pairs.
9	Temporal input pattern	Step, ramp, pulse, sine histories and queries; identical future plans across matched terminal states.
10	Observation corruption	Offset, scale, low-frequency drift, colored noise, soft clip; one segment covering 12.5/25/50% of historical PV.
11	Physical load	Load-strength coefficients $0.25/0.5/1$ and a clean reference; measured/hidden DV views of the same disturbed truth.
12	Integration	$K_I \in [0.0018, 0.0022]/[0.0045, 0.0055]/[0.009, 0.011]$ with dead time; sustained input imbalance has no finite steady state.

Intended use and coverage. The benchmark tests prescribed-input predictors and their use within the declared MPC procedure. The 1,024-step history and 96-step horizon define a common input contract; they are not tuned to each family's settling time or claimed to be optimal. Table 5 makes the

targeted coverage explicit. The protocol measures finite-window behavior, including slow responses that have not settled. It does not estimate the prevalence of these mechanisms in industrial plants.

Table 5: **Factor coverage in the formal evaluation.** "Core" denotes the registered core-control module. Stress controls use selected groups; prediction uses the full registered module.

Factor	Prediction families	Executed control coverage
Time constant, delay, gain	FOPDT	Core, all three levels
Mixing / outflow nonlinearity	Mixing / outflow	Core, all three levels
Damping / integration	SOPDT / integrating	Core, all three levels
Controller history	FOPDT	Three PI histories in core; nominal open-loop reference separately
Numerical input extrapolation	FOPDT, mixing, outflow, SOPDT	No dedicated paired extrapolation module
History–query patterns	FOPDT, mixing, outflow, SOPDT	Four history patterns; tracking targets rather than the prediction query patterns
Observation corruption	FOPDT, mixing, outflow, SOPDT	Soft clipping at three coverage levels, plus clean history
Physical load	FOPDT, outflow, SOPDT	Dose 0.5 histories; hidden/measured views and event/no-event pairs

B.1 Which dynamics generate the responses?

For intuition, the zero-load continuous-time family equations can be expressed using deviations x and Δu from a nominal operating point:

$$\text{FOPDT:} \quad \tau\dot{x} + x = K\Delta u(t-\theta), \tag{10}$$

$$\text{SOPDT:} \quad \ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = K\omega_n^2\Delta u(t-\theta), \qquad \omega_n = 1/\tau, \tag{11}$$

$$\text{Integrating:} \quad \dot{x} = K_I\Delta u(t-\theta). \tag{12}$$

For FOPDT at equilibrium, a step gives $x(t) = K\Delta u(1 - e^{-(t-\theta)/\tau})$ after the delay. This explains the separate roles of gain, time constant, and dead time. SOPDT instead permits overshoot, and integration has no finite equilibrium under a nonzero constant input imbalance.

The nonlinear families retain physical concentration or level states. The mixing tank satisfies $V\dot{c} = F_a c_a - (F_a + F_w)c$, where manipulated feed flow F_a is delayed and $\eta = F_{a0}/(F_{a0} + F_w)$. The outflow tank satisfies $A\dot{h} = u(t-\theta) - c_o h^\gamma$. Recorded PV is an affine function of concentration or level, with nominal scaling chosen for the benchmark comparisons. Thus a change in operating point can alter both response size and speed. These equations explain the mechanisms; numerical scoring uses the saved sampled trajectories, complete states, load terms, and independently checked simulator implementation, rather than fitting these equations to a model prediction.

Grouping and reuse. The physical base contains 100 FOPDT anchor instances, six sets of 100 matched parameter groups (time constant, delay, gain, mixing, outflow, damping), and 300 separately sampled integrating instances: 1,000 sampling groups and 2,200 parameterized process instances. The three levels in a matched group share their group identity; the integrating levels are independently sampled rather than treated as matched triples. Subsequent historical, observation, and information conditions reuse these registered process instances. Registered task identities do not imply that every numerical input array is unique. For example, the primary four-pattern module contains 3,200 registered histories but 3,043 distinct historical PV/MV arrays. All derived conditions retain their source-group identity.

Queries and matching. Standard conditions have one hold branch and six feasible action branches. The core completion procedure preserves legal original queries and records the actual direction and amplitude of replacements for infeasible slots. Four-pattern conditions have one hold and eight action branches, with two actions per input pattern. Their terminal segments are selected from physical-state convergence, independently of model performance. Across the primary four-pattern data, maximum state and future-output discrepancies are 9.9997×10^{-6} and 9.9831×10^{-6}, respectively. Different histories retain different excitation information even when terminal states match.

B.2 WHICH CONSTRUCTION RULES DETERMINE TASK DIFFICULTY?

Action amplitudes and feasible replacements. Let $a_u = (\max u - \min u)/2$ over the observed history and let u_h be its last input. Standard step queries hold $u_h \pm r a_u$ for 96 steps, with $r \in \{0.5, 1, 2\}$, alongside the hold query. Inputs must lie in $[0, 100]$, with additional positive-flow restrictions below; no separate step-size cap is applied. Actual historical minima and maxima determine inside/outside labels. Infeasible controller-history slots occur in FOPDT: legal original queries are retained, then the midpoint of the largest gap among existing target values and the endpoints 0 and 100 replaces each missing slot. Ties choose the lower gap. These 204 replacements retain their realized direction and amplitude and are excluded from fixed-amplitude-ratio groups. The implementation is `experiment/complete_formal_queries.py`; it does not replace cases according to model scores.

Observation corruption. Let $s = Q_{.95}(y) - Q_{.05}(y)$ and $c = \text{median}(y)$ for the clean history. One segment of length $L \in \{128, 256, 512\}$ starts at $\lfloor q(1024 - L + 1) \rfloor$, where the registered seeded draw $q \in [0, 1)$ is shared across coverage levels for that history. Within the segment, $y'_j = y_j + w_j \delta_j$; outside it, values are unchanged. The window w is one except for a half-cosine taper over approximately 10% of the segment at each internal boundary. Boundaries coinciding with the history endpoints are not tapered. Table 6 gives the fixed strength used in the formal module. MV, physical state, future plans, and future truth remain unchanged. Seeds and spans are stored per task by `experiment/formal_batch3.py`, which calls `experiment/local_corruption_pilot.py`.

Table 6: **Formal historical-PV corruption rules.** A seeded sign $\sigma \in \{-1, 1\}$ is used where applicable; $j = 0, \ldots, L - 1$. The common taper is applied after forming δ_j.

Kind	Perturbation before taper
Offset	$\delta_j = 0.1\sigma s$
Scale	$\delta_j = 0.25\sigma(y_j - c)$
Drift	$\delta_j = 0.1\sigma s\,[2j/(L-1) - 1]$
Colored noise	$\delta_j = 0.1 s z_j$, where $z_0 \sim \mathcal{N}(0,1)$ and $z_j = 0.5 z_{j-1} + \sqrt{0.75}\epsilon_j$, $\epsilon_j \sim \mathcal{N}(0,1)$ independently
Soft clip	$\delta_j = c + 0.5s \tanh[(y_j - c)/(0.5s)] - y_j$

Physical loads. In FOPDT and SOPDT, replace the delayed input deviation in the equations above by $u(t - \theta) - u_{\text{nom}} + v(t)$; in outflow, add $v(t)$ to delayed inflow. The load itself is not delayed. Histories use $v = \alpha a_u z$, with $\alpha \in \{0, 0.25, 0.5, 1\}$ and seeded plateau levels $z \in \{-1, -0.5, 0, 0.5, 1\}$. After a 256-step zero-load warmup, the observed history has eight 128-step plateaus, with a distinct final-sample level; fixed-query truth holds that final load constant. Measured views expose the load; hidden views use the same disturbed physical truth without that channel. Formal control uses dose-0.5 histories and holds the target at the final observed PV. The event condition adds $\pm 0.5 a_u$ to the final load from the first executed interval; its matched reference adds zero. The first decision receives no preview. See `experiment/formal_batch4.py`, `experiment/generate_registered_disturbance.py`, and `experiment/control_formal.py`.

Input domains, physical limits, and tracking targets. Control MV lies in $[0, 100]$. Mixing additionally requires $u > u_{\text{nom}} - F_{a0} = 38$; outflow requires positive inflow, and its lower MV bound under loads is tightened so that the registered load envelope preserves positive total inflow. Strict bounds use the next representable admissible value. FOPDT and SOPDT have no finite physical PV bounds. Integrating PV lies in $[0, 100]$. For mixing, $c_a = F_{a0}/[\eta(1 - \eta)K]$ and the registered PV limits are $y_{\text{nom}} - \eta c_a K$ and $y_{\text{nom}} + (1 - \eta) c_a K$. For outflow, the lower PV limit is $y_{\text{nom}} + (10^{-9} - y_{\text{nom}}) \gamma u_{\text{nom}} K / y_{\text{nom}}$, corresponding to the minimum height; there is no finite upper bound. These physical limits are distinct from the common numerical $|\hat{y}| < 1000$ candidate check.

For stable-family tracking, let $[L_y, U_y]$ be the equilibrium-output range over legal MV, intersected with physical PV limits. For integration, use the physical PV limits alone. From final historical PV y_h, set

$$\delta_y = \tfrac{1}{2} \min\{s_y, y_h - L_y, U_y - y_h\}, \qquad r = y_h \pm \delta_y,$$

omitting an unbounded margin. The task constructor uses dynamics for this target selection; learned predictors receive the target and public domains, not plant parameters or hidden state. One target sign

is assigned per sampling group by alternating signs in a fixed SHA-256 group ordering and is shared across conditions. Load tasks instead retain $r = y_h$. Pattern controls share the step-history target and scales; corrupted-history controls use the clean-history target and scales. These rules are implemented in `experiment/control_development.py` and `experiment/control_formal.py`.

From instances and histories to control tasks. The 2,200 physical instances yield two original histories each (4,400). Core control uses seed zero: 1,800 histories from six matched physical axes, 300 integrating histories, and three PI histories for each of 100 FOPDT anchor groups, totaling 2,400 tasks before development exclusion. Another 100 nominal FOPDT tasks form a separate module. Stress controls use the first 80 hash-ranked groups in each eligible family: four families for patterns/corruption, three for loads. Removing the same 70 development groups gives the module counts in Table 12. This is a declared selection of conditions, not a new sample of independent processes.

Reference validation and applicable windows. Physical references are checked against analytical or independent numerical solutions. Maximum recorded reference error is 7.13×10^{-10} in the first physical-data batch and 2.49×10^{-11} in the four-pattern module. These numerical checks establish implementation agreement within the specified models; they are not external validation against industrial plants. Full-horizon scores retain valid slow and integrating responses. Step-response phases are diagnostic and can be absent; the four-pattern module uses full-horizon scores without imposing step-response phases on ramps, pulses, or sinusoids. Observation and information views do not create new independent physical responses.

Realized finite-horizon coverage. Table 7 aggregates the saved window audits of the original physical base. Each of its 2,200 process instances contributes two histories and six non-hold step actions per history; each registered history–action pair is counted once. Later histories, observation corruptions, and DV views are excluded from these counts. Thus, the denominator is 26,400 physical responses, not the 135,700 queries in the expanded inventory, and is not a count of independent systems. Settling is the first index after which the true incremental response d_a remains within a 2% band around its equilibrium effect in the independently checked long reference continuation. A post-transient window exists only if the settling index precedes the end of the 96-step horizon. A finite settling index at or beyond the horizon is recorded as truncated; integrating responses have no finite equilibrium effect or settling time. Unresolved cases are a separate status, with zero occurrences in this base. The 624 truncated SOPDT responses all occur at $\zeta = 0.3$ (624/1,200); outflow contributes 351/1,200 at $\gamma = 0.1$ and 43/1,200 at $\gamma = 0.22$; mixing contributes 29/1,200 at $\eta = 0.9$. Other non-integrating strata have no truncated responses.

Table 7: **Observed response coverage in the physical base.** Counts are non-hold history–action pairs. All three status columns remain eligible for full-horizon scoring; missing post-transient windows do not indicate failed data generation.

Process family	Responses	With post window	Truncated	No finite settling
FOPDT	12,000	12,000	0	0
Mixing	3,600	3,571	29	0
Outflow	3,600	3,206	394	0
SOPDT	3,600	2,976	624	0
Integrating	3,600	0	0	3,600
Total	**26,400**	**21,753**	**1,047**	**3,600**

Figure provenance. Panel (a) of Figure 2 shows the saved 1,024-step PV/MV history, hold plan, and two signed step plans with their 96-step physical outputs for the FOPDT instance in panel (b). Panel (b) uses instance index zero and history seed zero from the FOPDT anchor, mixing $\eta = 0.9$, outflow $\gamma = 0.1$, SOPDT $\zeta = 0.3$, and middle integrating-gain strata. The positive and negative queries have magnitude twice the historical MV half-range. Each pair in panel (b) is divided by its common maximum absolute action–hold response to expose shape without changing its directional asymmetry. This fixed selection illustrates mechanisms and does not estimate their population frequency. Panel (c) summarizes the parameter settings in Table 4, separating mixing and outflow as evaluation factors and excluding the FOPDT anchor as a separate factor. Fast-to-slow, short-to-long, and low-to-high labels denote ordered parameter ranges, not model-independent difficulty. The

portable figure data retain source task IDs and file hashes; full response-window counts remain in Table 7.

C CURRENT MODEL AND CONTROL PROTOCOL

Scope of this supplement. This supplement documents the completed five-family simulation and control evaluation and its clean-FOPDT paired analysis. The twelve construction components in Appendix B include a separate FOPDT anchor and a combined nonlinear component. The twelve result panels in Figure 7 instead separate mixing and outflow nonlinearity. Neither list denotes twelve state variables or a fully crossed experimental design.

Evaluation populations and run audit. Each model processes 17,100 conditions and 135,700 queries, including 17,100 hold references and 118,600 non-hold actions. Main scores preserve the frozen evaluation-cell weighting. The physical-core diagnosis instead uses 4,400 histories and their six original step actions, giving 26,400 action/hold pairs per model, with errors and reference magnitudes averaged within and then across groups. These populations and weights are kept separate. Parameter and condition contrasts use 2,000 group bootstrap replicates; paired conditions are resampled together, while independently sampled integrating levels are resampled separately. These intervals are exploratory and unadjusted for multiplicity. Full-inventory aggregate simulation scores are point estimates; group confidence intervals for that leaderboard have not been computed.

Control populations and search budget. The registered control inventory contains 6,020 tasks per configuration over 1,000 groups. Seventy development or previously inspected control groups are excluded from primary testing, leaving 930 test groups and 5,594 tasks per configuration. The test inventory comprises 2,228 core-tracking tasks, 94 nominal FOPDT references, 1,188 history-pattern tasks, 1,188 corruption tasks, and 896 load tasks. All 54,180 runs across nine configurations were independently audited: 54,056 completed the full horizon and 124 terminated physically, with no missing runs or infrastructure failures. This split isolates control development groups; it is not a claim that the associated simulation results were previously unseen.

History-only model selection. The supplied history has 1,024 observations. Local baselines use rolling-origin validation within that history; neither future query truth nor closed-loop test outcomes enter fitting or configuration selection. Training-segment statistics determine standardization. ARX includes output feedback; the tested FIR, MLP, and RBF-SVR use finite input histories, which can restrict their treatment of accumulation. Measured-DV views fit their corresponding information interfaces rather than borrowing models fitted without DV. Control reuses each context/view's saved local-model parameters. TSFM bindings and historical-version limitations are specified below. No model is retrained during a control episode. Numerical protections are distinct from physical-domain termination. Zero failed final simulation predictions does not imply that every candidate fit was valid or that every prediction is accurate.

C.1 LOCAL BASELINES: REPRESENTATIONS, FITTING, AND SELECTION

Input features and prediction targets. All four local baselines predict PV levels rather than output increments. With the stored sample index t and input order q, their input-history vector contains $(u_{t-1}, \ldots, u_{t-q})$. Measured-DV models additionally receive $(v_t, \ldots, v_{t-q+1})$; clean and hidden-DV models have no DV features. Thus the input-feature dimension is $d = q$ or $2q$, respectively. ARX prepends n_y previous PV values, while FIR, MLP, and RBF-SVR do not use PV as a prediction feature. Historical PV still supplies their training targets. Setpoints and physical process parameters are excluded. At forecast time, these windows combine observed inputs with the proposed MV plan and, where available, the hold-last DV continuation.

ARX and FIR. Both fit an affine ridge regression to standardized lag features. Writing ϕ_t for the corresponding feature vector and $z_t = (\phi_t - \mu)/\sigma$, the fitted predictor is $\hat{y}_t = \bar{y} + z_t^\top \beta$. The statistics μ, σ, and $\bar{y}$ are computed on valid lagged training rows and held fixed when forecasting; a zero feature standard deviation is replaced by one. Coefficients minimize mean squared error plus $\lambda\|\beta\|_2^2$, with an unpenalized intercept. The implementation uses singular-value decomposition, with a

recorded SciPy SVD fallback if the NumPy decomposition fails. ARX recursively feeds its predicted PV into subsequent forecast steps. FIR uses only the finite input window, so it has no analogous output-state recursion. These are history-fitted predictors, not fits to known plant coefficients.

MLP and RBF-SVR. The MLP maps the same input-only lag vector through one ReLU hidden layer and a scalar output layer. RBF-SVR uses an ϵ-insensitive support-vector regressor with kernel $k(z, z') = \exp(-\gamma\|z - z'\|_2^2)$. Both use scikit-learn pipelines that standardize the features and the PV target using the training prefix, then invert the target transformation for prediction. Neither feeds predicted PV back into its input features. The SVR bandwidth grid is $\gamma \in \{0.1, 1, 10\}/d$; the $\epsilon = 0.01$ tolerance is in standardized target units. Thus measured-DV fits adjust the kernel scale for their doubled input-feature dimension.

Hyperparameter search. Each local baseline is selected separately for each supplied history and information view, using the search spaces in Table 8. Let n_y denote the PV lag order, q the MV lag order, and d the input-feature dimension excluding PV lags. We use $d = q$ for MV-only histories and $d = 2q$ when measured DV is included. Define $\mathcal{Q} = \{16, 32, 64, 96, 128, 192\}$ and $\Lambda = \{10^{-8+7j/8} : j = 0, \ldots, 8\}$.

Table 8: **Local-baseline search spaces.** Candidate counts are per history and information view. They count hyperparameter configurations, not individual validation fits.

Model	Search space	Candidates
ARX	$n_y \in \{2, 4, 8, 16, 32\}$; $q \in \{16, 24, 32, 64\}$; ridge penalty $\lambda \in \Lambda$.	180
FIR	$n_y = 0$; $q \in \mathcal{Q}$; ridge penalty $\lambda \in \Lambda$.	54
MLP	$n_y = 0$; $q \in \mathcal{Q}$; one ReLU hidden layer of width 32 or 64; regularization $\alpha \in \{10^{-4}, 10^{-2}\}$.	24
RBF-SVR	$n_y = 0$; $q \in \mathcal{Q}$; $C \in \{1, 10, 100\}$; $\gamma \in \{0.1, 1, 10\}/d$; fixed $\epsilon = 0.01$ in standardized target units.	54

Historical validation. Every candidate is evaluated using five expanding-window folds within the supplied 1,024-step history, each with a 96-step validation horizon. The frozen task manifest specifies the forecast origins: $(544, 640, 736, 832, 928)$ for core and observation-corruption histories, $(385, 513, 641, 769, 897)$ for load histories, and $(256, 352, 448, 544, 640)$ for the four-pattern histories. Preprocessing statistics are fitted on the training prefix of each fold. The held-out historical MV supplies the validation input plan; measured DV is held at its last training-prefix value throughout the validation horizon. ARX recursively uses its predicted PV during each forecast. Neither the evaluation-query truth nor subsequent closed-loop outcomes enter model selection.

Selection and refitting. A candidate is eligible only if all five folds produce finite predictions with absolute values below 1,000. We select the lowest mean validation RMSE. Exact ties are resolved by the smaller feature dimension, followed by the larger ridge penalty for ARX/FIR or the earlier search-grid entry for MLP/RBF-SVR. If no candidate is eligible, the fit is recorded as failed. The selected configuration is refitted on the valid lagged rows of the full 1,024-step history. Its fitted parameters and preprocessing statistics are saved and reused for prediction and control from the corresponding history/view. During control, observations update the input history while the fitted predictor remains fixed.

Solver settings and fit records. MLP uses L-BFGS with at most 10,000 iterations and 500,000 function evaluations, tolerance 10^{-4}, and random seed 20260922. RBF-SVR uses tolerance 10^{-3}. The fit records retain candidate configurations, fold errors, validity status, convergence warnings, and the selected configuration. Convergence warnings do not automatically disqualify an otherwise valid candidate. Consequently, successful final predictions do not imply that every attempted candidate converged.

C.2 TSFM BASELINES: CHECKPOINTS, ADAPTERS, AND INFERENCE

All three TSFMs receive 1,024 historical PV values and aligned historical/planned MV; measured-DV conditions add DV after MV, holding the latest observed DV over the future horizon. Setpoints are excluded. Forecast indices immediately follow the observed history, with no adapter-applied temporal

shift. Adapters pass raw physical values as float32 without benchmark-level standardization and return float64 arrays; model-internal normalization is delegated to the SDK. The requested horizon is 96, with batch size 8 for fixed-query prediction and 128 for control.

Table 9: **Public pretrained checkpoints and forecast extraction.**

Model	Checkpoint	Adapter and point output
Chronos-2	`amazon/chronos-2`	Historical PV target; named historical/future covariates; requested 0.5 quantile.
TiRex-2	`NX-AI/TiRex-2`	Historical PV target; concatenated past/future covariates; median (quantile index 4).
TimesFM-3	`google/timesfm-3.0-pytorch`	Historical PV target; concatenated past/future covariates; SDK point output, symmetric averaging disabled.

Chronos-2. The adapter loads the directory containing the recorded `model.safetensors` file through `Chronos2Pipeline.from_pretrained`. Historical PV is the target; MV and optional DV are named `channel_0` and `channel_1` covariates, with separate historical and future segments. Inference calls `predict_quantiles` with `quantile_levels=[0.5]`. The returned point trajectory is therefore a median, even where internal helper names use the word "mean." No task-specific fine-tuning is performed.

TiRex-2. Historical PV is supplied as a $(1, 1024)$ target array. The adapter sets `past_covariates=None`; the SDK's `future_covariates` argument instead receives concatenated historical and planned covariates of shape $(C, 1120)$, where $C = 1$ for MV and $C = 2$ with measured DV. Quantile index 4 supplies the median trajectory. The control adapter additionally caches history-only trend detection, so its source hash differs from the fixed-query adapter. A six-episode, 16-step development check found unchanged forecasts, plans, and actions; it does not establish equivalence over all formal tasks.

TimesFM-3. The adapter likewise supplies historical PV and concatenated past/future covariates. It requests `return_quantiles=False` and `use_symmetric_averaging=False`, then extracts the SDK point forecast. The run records call this output `model_point`; we do not relabel it as a mean or median without historical SDK evidence. The stored checkpoint directory includes both weights and model configuration.

Historical checkpoint identity. Table 10 identifies the weight and model-configuration files recorded for fixed-query simulation. These are historical byte identities, not hashes newly measured from current downloads. A checkpoint's public name alone is not a revision pin. Chronos records its weight file but not the complete directory metadata used by its loader. Formal control records retain checkpoint paths, configurations, and frozen adapter sources, but do not independently record checkpoint byte hashes. Consequently, matching paths cannot establish identical historical weights between prediction and control.

Table 10: **Recorded simulation checkpoint files.** Hashes show the first 16 hexadecimal characters of SHA-256 for readability; the complete values are in the accompanying configuration JSON. No row is an independently recorded control-weight hash.

Model	File	SHA-256 prefix
Chronos-2	`model.safetensors`	`ddcda3c7508bf2528`
TiRex-2	`model.ckpt`	`184b160ffbe4c01a`
	`model-config.yaml`	`cddbe6d0be4919cb`
TimesFM-3	`model.safetensors`	`a7592b0a8432baee`
	`config.json`	`ff17bbc07b792c5a`

Dependency versions and remaining gaps. Table 11 separates recoverable version evidence from unrecorded fields. The run archives do not provide a complete historical SDK/dependency lock or checkpoint repository revisions. Current runtime packages and current installation requirements describe a later environment and cannot replace that evidence. The reported float32 inputs and float64 outputs do not establish all SDK-internal precision or determinism settings. These limits

qualify the paired analysis as a comparison of recorded configurations, rather than a fully verified identical-runtime transfer test.

Table 11: **Formal-run dependency evidence.** "Not recorded" indicates that the inspected archives do not establish the package version used in the full paper runs. Environment records from earlier pipeline checks, later interface validation, or the current installation are not used to fill these fields. The reported scikit-learn version is supported by the two inspected fitted artifacts and does not establish a complete environment lock for all runs.

Dependency	Recoverable evidence and scope
Python	3.12.12 in simulation provenance; no separately bound control version.
NumPy / SciPy	Historical run-bound versions not recorded.
scikit-learn	1.9.0 in the two inspected, hash-matched MLP/SVR fitted artifacts from one registered task; not an inventory-wide environment lock.
PyTorch / Transformers	Historical TSFM run-bound versions not recorded.
`chronos-forecasting`	Historical package version not recorded.
TiRex-2 / TimesFM-3 SDKs	Historical package versions and upstream revisions not recorded.
`safetensors`	Historical package version not recorded.

C.3 Model bindings from prediction to control

Local fitted models. A local predictor is identified by model name, source task/history, and information view, not by model class alone. Its saved artifact contains the fitted estimator or ridge coefficients, preprocessing statistics, selected lag orders and hyperparameters, and historical-validation report. The prediction record hashes both the prediction array and the fitted artifact. Each registered control task binds its source input, physical reference, and corresponding model artifact by SHA-256. Before execution, the runner checks these identities and the artifact's PV target and MV/DV input-channel contract. A changed artifact raises an error rather than triggering an unrecorded refit. Observations update the rolling history during execution, while fitted coefficients, scalers, and hyperparameters remain frozen.

Frozen TSFM runtimes. The TSFM binding contains the checkpoint path/name and effective inference configuration, including context length, batch size, device, and interface options. Adapter source identities are also retained. Chronos-2 and TimesFM-3 use matching adapter-file hashes across the two workflows; TiRex-2 has the separate control-cache version described above. The simulation weight hashes, missing independent control-weight hashes, and incomplete dependency records remain distinct fields. Reusing a frozen runtime within a control episode does not, by itself, prove its historical identity with the earlier simulation runtime.

Controller references and binding records. The PI reference binds a saved MV-only ARX surrogate and its tuning history; both load-information views use the hidden-DV surrogate without feedforward. True-model MPC uses the registered physical dynamics and current internal state rather than a learned-model artifact. Their execution rules follow below. The accompanying `iclr/reproducibility_config.json` records the effective baseline settings, full checkpoint/adapter hashes, a representative local-artifact binding, and explicit missing fields. This recovered configuration record does not replace the complete task-level registry or a historical environment lock.

C.4 How is closed-loop control executed?

Control inventory and overlap. Table 12 separates the test inventory from development. Counts in the group column overlap across modules; their union is 930. The full registered inventory has 1,000 groups, 6,020 tasks per configuration, and 54,180 runs for nine configurations. Each group has one preassigned target direction, shared across matched levels and views. The full inventory is balanced at 3,010 positive and 3,010 negative targets per configuration. Physical domains and legal targets are defined at task construction; these constraints do not supply hidden dynamics to the learned predictor. Historical scales and legal targets can differ across parameter levels, so those comparisons do not isolate a single causal parameter effect.

Table 12: **Registered control populations per configuration.** Common-complete counts refer to the intersection across all nine configurations. Group counts must not be summed.

Module	All tasks	Test tasks	Test groups	Common complete
Core tracking	2,400	2,228	930	2,166
Nominal FOPDT	100	94	94	94
Historical pattern	1,280	1,188	297	1,173
Historical soft clipping	1,280	1,188	297	1,182
Physical load	960	896	224	894
Total	6,020	5,594	930 (union)	5,509

Control information and execution. MPC uses the public input domain and eight action blocks of lengths $(1, 1, 2, 4, 8, 16, 32, 32)$. It scores 32 candidates in each of two CEM rounds, retains eight elites, shifts a warm-start plan by four steps, and executes the first four actions before replanning. Physical feedback and boundary checks occur at each execution step. Sampling uses a truncated normal in the normalized legal action domain, initial standard deviation 0.5, an update with 0.9 elite weight, and standard-deviation floor 0.01. Explicit candidates include the current hold, shifted warm start, and best-so-far plan, with duplicates removed. The first decision starts from hold. A shared NumPy seed is derived from the first eight SHA-256 bytes of `20260924:pair_id:direction:step:0`; it excludes model identity. Subsequent sampled plans can differ as model-specific elites and warm starts change.

A candidate receives infinite cost if a predicted PV is nonfinite, has absolute value at least 1,000, or leaves the registered task-specific output domain. CEM uses finite-cost candidates only, with up to eight elites. Output upper bounds are inclusive; lower bounds are inclusive at zero and strict otherwise. Actions obey the registered MV bounds; the objective penalizes input changes without imposing a separate hard slew-rate constraint. If neither CEM round finds a finite-cost candidate, the controller holds the last applied MV until the next scheduled four-step replan. This fallback does not certify feasibility of the held input. Runtime exceptions instead enter job retry/error handling, not numerical fallback. Nonfinite actual PV or a physical-domain violation terminates execution.

Fallback episodes contain at least one fallback step. Step counts count executed actions, while decision counts count replanning events: one fallback decision normally contributes four steps, unless execution terminates earlier. In an archived outflow trace, all 64 candidates fail at step 8, steps 8–11 hold the step-7 input, and normal planning resumes at step 12. This is distinct from physical termination. The true-model reference has the true internal state and current DV in both load views, but no knowledge of a new future load event. Hence its comparison with hidden-DV learned models includes information privilege as well as model fidelity.

PI is tuned on the frozen MV-only ARX surrogate using a 48-point grid. The normalized proportional gains are 0.0625, 0.125, 0.25, 0.5, 1, 2, 4, and 8; integral times are 1, 4, 16, 64, 256, and 1,024. Three historical starting points and two target directions provide six 96-step tuning rollouts. PI observes feedback every step and has no DV feedforward; both load views use the same hidden-DV ARX surrogate and history for tuning. A PI tuning candidate must yield valid costs on all six surrogate rollouts. If none does, no PI controller is instantiated: the episode holds its last input throughout, without retuning. PI fallback counts denote such episodes, rather than failure of a CEM search; their per-step flags do not represent repeated tuning attempts.

The load event occurs in the first executed action interval, without advance notice at the first decision. Measured-DV models subsequently receive observed DV and its hold-last continuation; hidden-DV models receive no DV. Event and no-event counterparts share the first decision seed. Historical corruption affects the model's past observations and corresponding fit; future executed feedback is clean. This is a history-corruption recovery test, not persistent sensor corruption during execution. Pattern-history tasks preserve their actual terminal states and share the step-history target and scales; they are not terminal-state-matched causal tests of history alone.

Aggregation and uncertainty. Simulation main scores first average each raw error and reference magnitude within a declared evaluation cell, then weight cells equally, then form skill as in Eq. (4). Physical-core simulation diagnostics average within group and then across groups, preserving all three independent integrating strata. In control, core conditions are averaged within group, followed by group means within axes/strata, four FOPDT axes equally, three integrating strata equally, and

finally the five physical families equally. Stress modules have their own family/group/condition weights and are not merged into one global score. Physical terminations retain their original weights for full-denominator measures; a failed short trajectory does not enter a 96-step error mean.

Core control inference resamples complete groups within registered strata, retaining all conditions and model columns together. Percentile intervals describe paired sampling uncertainty. Six comparisons against ARX use Holm correction with two-sided centered-bootstrap probabilities

$$p_{\text{boot}} = \frac{1 + \sum_{b=1}^{B_{\text{ctrl}}} \mathbf{1}\{|\Delta_b^* - \Delta| \geq |\Delta|\}}{B_{\text{ctrl}} + 1}.$$

Here, Δ is the observed difference, Δ_b^* is its value in bootstrap replicate b, and $\mathbf{1}\{\cdot\}$ is the indicator function. We use $B_{\text{ctrl}} = 10{,}000$ replicates. This formula was specified during statistical implementation, not fully before the experiment. Family and stress intervals are exploratory and do not have simultaneous familywise coverage. Simulation contrasts use 2,000 group replicates. The registered sampling distribution is synthetic; neither bootstrap scheme estimates industrial prevalence.

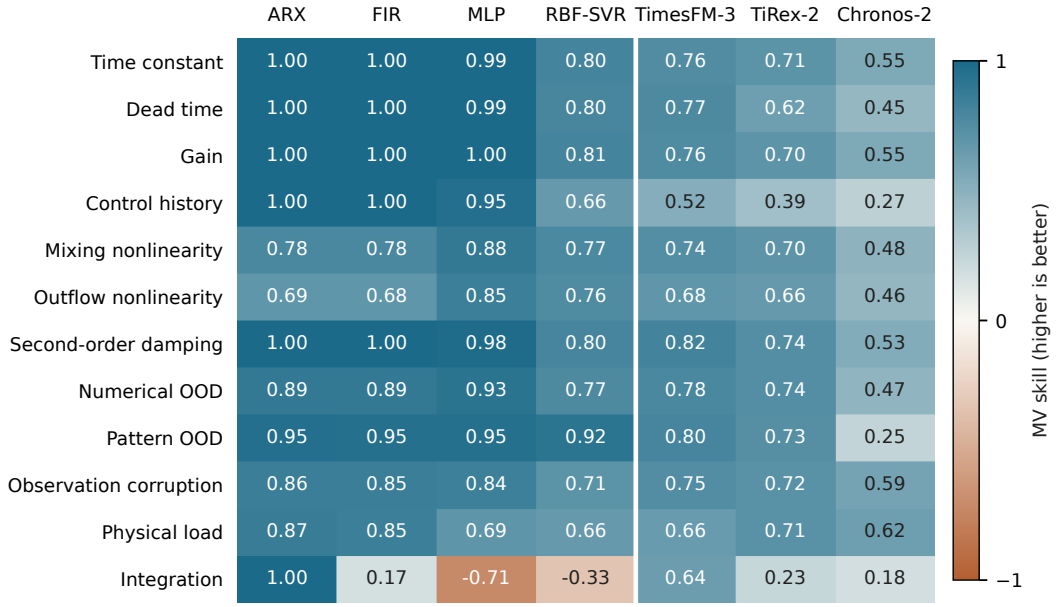

Figure 7: **All twelve simulation panels.** MV skill with full-inventory evaluation-cell weighting. Panels can share reference data and are not disjoint samples. Values are point estimates on a common scale; negative values are retained.

Parameter sensitivity. In the physical-core delay contrast, TimesFM-3, TiRex-2, and Chronos-2 have long-minus-short MV-skill changes of -0.278 ($[-0.296, -0.261]$), -0.276 ($[-0.300, -0.253]$), and -0.493 ($[-0.521, -0.466]$), respectively. These are exploratory 95% group intervals. Gain changes instead mainly rescale response and raw error: all three high-minus-low skill changes are below 0.003 in magnitude. Figure 4 uses the physical-core parameter slices and the corresponding group-equal controller-history slice, not the full-panel weighting. Full level and contrast values are retained in the figure-data snapshot.

D Simulation detail: physical levels and stress conditions

D.1 Do the aggregate scores hide different physical regimes?

Table 13 retains every level of the physical core. Its 22 strata each contain 100 groups, two histories per group, and six original non-hold step queries per history. Parameter levels within the six stable-process axes are matched; the three integrating levels use independently sampled groups. Errors and reference magnitudes are averaged within groups and then across groups before skill is computed.

The nonlinear rows explain why one linear-model score is insufficient. As mixing and outflow nonlinearities increase, locally fitted linear predictors lose response accuracy, whereas MLP retains

Table 13: **Complete physical-core MV skill.** Levels 1–3 for gain, time constant, dead time, and integration follow the ranges in Table 4. Nonlinear and damping rows show the actual parameter value.

Factor	Level	ARX	FIR	MLP	RBF-SVR	TimesFM-3	TiRex-2	Chronos-2
Anchor	Anchor	1.000	1.000	0.995	0.807	0.770	0.711	0.569
Time constant	1	1.000	1.000	0.990	0.801	0.726	0.676	0.552
Time constant	2	1.000	1.000	0.995	0.812	0.754	0.703	0.556
Time constant	3	1.000	1.000	0.993	0.801	0.795	0.753	0.540
Dead time	1	1.000	1.000	0.995	0.810	0.898	0.698	0.610
Dead time	2	1.000	1.000	0.994	0.802	0.758	0.701	0.547
Dead time	3	1.000	1.000	0.991	0.772	0.620	0.422	0.117
Gain	1	1.000	1.000	0.995	0.806	0.760	0.700	0.552
Gain	2	1.000	1.000	0.995	0.805	0.762	0.700	0.553
Gain	3	1.000	1.000	0.995	0.805	0.762	0.700	0.553
Mixing	0.2	0.918	0.918	0.959	0.818	0.751	0.708	0.553
Mixing	0.5	0.793	0.793	0.901	0.803	0.720	0.690	0.533
Mixing	0.9	0.613	0.613	0.793	0.745	0.632	0.621	0.473
Outflow	0.85	0.983	0.983	0.989	0.813	0.761	0.704	0.543
Outflow	0.22	0.690	0.687	0.856	0.788	0.687	0.660	0.514
Outflow	0.1	0.401	0.396	0.737	0.736	0.551	0.554	0.421
Damping	0.8	1.000	1.000	0.995	0.823	0.815	0.679	0.481
Damping	0.5	1.000	1.000	0.992	0.831	0.834	0.759	0.608
Damping	0.3	1.000	1.000	0.977	0.806	0.794	0.744	0.683
Integration	1	1.000	0.151	-0.721	-0.432	0.635	0.258	0.186
Integration	2	1.000	0.131	-0.720	-0.321	0.634	0.257	0.180
Integration	3	1.000	0.193	-0.706	-0.317	0.638	0.213	0.175

an advantage. Integration produces a different ordering: FIR and MLP do not track accumulated state as accurately as the tested TSFMs. These mechanisms should not be collapsed into a common "hardness" scale.

Table 14: **Output-trajectory skill for all twelve panels.** These are full-inventory, cell-weighted scores and complement the MV-skill matrix in Figure 7. They are not the group-weighted physical-core scores above.

Panel	ARX	FIR	MLP	RBF-SVR	TimesFM-3	TiRex-2	Chronos-2
Time constant	1.000	1.000	0.993	0.816	0.773	0.729	0.558
Dead time	1.000	1.000	0.994	0.810	0.789	0.635	0.444
Gain	1.000	1.000	0.995	0.813	0.775	0.716	0.557
PI history	1.000	1.000	0.959	0.671	0.553	0.472	0.360
Mixing	0.802	0.797	0.893	0.791	0.757	0.723	0.489
Outflow	0.711	0.701	0.861	0.780	0.707	0.687	0.469
Damping	1.000	1.000	0.978	0.823	0.833	0.761	0.523
Numerical OOD	0.899	0.895	0.934	0.798	0.804	0.759	0.480
Pattern OOD	0.964	0.962	0.947	0.920	0.798	0.731	0.246
Corruption	0.864	0.852	0.845	0.721	0.761	0.733	0.593
Physical load	0.854	0.785	0.694	0.661	0.675	0.639	0.603
Integration	1.000	-0.572	-1.158	-0.794	0.656	0.256	0.197

D.2 What changes under historical corruption and input shifts?

Table 15 separates five corruption transformations and three coverage levels. The altered segment lies in the historical observations; future truth is unchanged. Increasing the affected fraction is not uniformly harder: scale, offset, drift, noise, and clipping remove or change different information. A model's improvement on one corrupted slice does not establish that corruption is beneficial generally.

Numerical extrapolation changes the future input's range relative to the observed history. Pattern transfer changes the relation between historical excitation and the future plan. These are distinct tests. In the pattern experiment, terminal states and future responses are approximately matched, so the comparison examines what the different histories allow the predictor to infer. Table 16 retains both score endpoints and paired intervals for numerical, pattern, information-view, and corruption comparisons; reference magnitudes are retained in the accompanying snapshot. Neither an OOD

Table 15: **All corruption types and coverages.** Group-weighted MV skill; each cell pools 400 groups from four non-integrating families. This table gives absolute scores; paired differences use the corresponding matched clean or lower-coverage condition.

Corruption	Coverage	ARX	FIR	MLP	RBF-SVR	TimesFM-3	TiRex-2	Chronos-2
Offset	12.5%	0.867	0.863	0.861	0.747	0.755	0.718	0.580
Offset	25%	0.867	0.852	0.817	0.660	0.750	0.723	0.586
Offset	50%	0.869	0.843	0.795	0.596	0.742	0.725	0.586
Scale	12.5%	0.864	0.863	0.874	0.747	0.756	0.718	0.580
Scale	25%	0.845	0.842	0.837	0.690	0.751	0.721	0.587
Scale	50%	0.808	0.802	0.793	0.618	0.738	0.717	0.581
Drift	12.5%	0.875	0.872	0.901	0.782	0.763	0.714	0.577
Drift	25%	0.872	0.865	0.882	0.757	0.761	0.717	0.580
Drift	50%	0.871	0.857	0.850	0.701	0.759	0.722	0.579
Colored noise	12.5%	0.873	0.871	0.865	0.801	0.731	0.722	0.623
Colored noise	25%	0.870	0.868	0.828	0.782	0.749	0.732	0.657
Colored noise	50%	0.867	0.864	0.794	0.755	0.752	0.742	0.695
Soft clip	12.5%	0.866	0.864	0.884	0.750	0.751	0.708	0.558
Soft clip	25%	0.850	0.847	0.838	0.693	0.739	0.702	0.549
Soft clip	50%	0.818	0.814	0.786	0.581	0.719	0.685	0.531

panel average nor a load-panel average alone is a degradation estimate, because each also contains reference conditions.

Table 16: **Paired input, history, and information comparisons.** Group-weighted MV skill at both endpoints and their paired difference. Conditions retain their original matching; intervals are exploratory.

Comparison	Model	Reference	Changed	ΔMV skill [95% CI]
Measured minus hidden	ARX	0.820	0.899	0.079 [0.071, 0.088]
Measured minus hidden	Chronos-2	0.567	0.674	0.108 [0.095, 0.120]
Measured minus hidden	FIR	0.772	0.898	0.125 [0.113, 0.138]
Measured minus hidden	MLP	0.365	0.928	0.563 [0.535, 0.591]
Measured minus hidden	RBF-SVR	0.368	0.870	0.502 [0.474, 0.533]
Measured minus hidden	TimesFM-3	0.602	0.661	0.059 [0.048, 0.070]
Measured minus hidden	TiRex-2	0.725	0.672	-0.052 [-0.062, -0.043]
Outside minus inside	ARX	0.949	0.905	-0.043 [-0.049, -0.037]
Outside minus inside	Chronos-2	0.722	0.604	-0.118 [-0.128, -0.109]
Outside minus inside	FIR	0.948	0.905	-0.042 [-0.048, -0.036]
Outside minus inside	MLP	0.983	0.962	-0.021 [-0.025, -0.018]
Outside minus inside	RBF-SVR	0.984	0.906	-0.077 [-0.083, -0.071]
Outside minus inside	TimesFM-3	0.927	0.831	-0.096 [-0.105, -0.087]
Outside minus inside	TiRex-2	0.892	0.806	-0.086 [-0.095, -0.079]
Corrupted minus clean	ARX	0.880	0.859	-0.021 [-0.023, -0.019]
Corrupted minus clean	Chronos-2	0.557	0.590	0.033 [0.029, 0.038]
Corrupted minus clean	FIR	0.879	0.853	-0.027 [-0.029, -0.025]
Corrupted minus clean	MLP	0.940	0.840	-0.100 [-0.105, -0.094]
Corrupted minus clean	RBF-SVR	0.810	0.711	-0.099 [-0.106, -0.092]
Corrupted minus clean	TimesFM-3	0.760	0.748	-0.012 [-0.016, -0.008]
Corrupted minus clean	TiRex-2	0.703	0.718	0.014 [0.011, 0.018]
Mismatched minus matched	ARX	0.955	0.954	-0.001 [-0.001, -0.001]
Mismatched minus matched	Chronos-2	0.414	0.191	-0.223 [-0.230, -0.217]
Mismatched minus matched	FIR	0.955	0.951	-0.003 [-0.004, -0.003]
Mismatched minus matched	MLP	0.981	0.935	-0.045 [-0.046, -0.045]
Mismatched minus matched	RBF-SVR	0.973	0.902	-0.071 [-0.072, -0.069]
Mismatched minus matched	TimesFM-3	0.881	0.770	-0.111 [-0.115, -0.107]
Mismatched minus matched	TiRex-2	0.877	0.682	-0.195 [-0.202, -0.187]

D.3 How large are the paired changes?

Table 17 reports all 112 physical, history, corruption-coverage, and load-information contrasts from the current group analysis. "Level 3–1" means the last minus first physical level in the declared ordering; for outflow it is $\gamma = 0.1$ minus 0.85, and for damping it is $\zeta = 0.3$ minus 0.8. A negative change denotes lower MV skill. Corruption contrasts compare 50% with 12.5% coverage, not with clean history. Load contrasts compare measured with hidden DV at the same load strength. Intervals are marginal exploratory intervals, not simultaneous evidence that every nonzero row generalizes.

Table 17: **Complete simulation contrasts.** Right minus left; group bootstrap intervals are exploratory and unadjusted. Integrating levels are independent; other rows retain matched groups.

Contrast	Model	ΔMV skill [95% CI]	Groups L/R
tau:level 3–1	ARX	0.000 [0.000, 0.000]	100/100
tau:level 3–1	FIR	-0.000 [-0.000, -0.000]	100/100
tau:level 3–1	MLP	0.003 [0.002, 0.004]	100/100
tau:level 3–1	RBF-SVR	0.000 [-0.008, 0.009]	100/100
tau:level 3–1	TimesFM-3	0.069 [0.060, 0.080]	100/100
tau:level 3–1	TiRex-2	0.078 [0.066, 0.090]	100/100
tau:level 3–1	Chronos-2	-0.012 [-0.025, 0.001]	100/100
theta:level 3–1	ARX	-0.000 [-0.000, -0.000]	100/100
theta:level 3–1	FIR	-0.000 [-0.000, -0.000]	100/100
theta:level 3–1	MLP	-0.004 [-0.005, -0.003]	100/100
theta:level 3–1	RBF-SVR	-0.039 [-0.047, -0.030]	100/100
theta:level 3–1	TimesFM-3	-0.278 [-0.296, -0.261]	100/100

Continued from previous page

Contrast	Model	ΔMV skill [95% CI]	Groups L/R
theta:level 3–1	TiRex-2	-0.276 [-0.300, -0.253]	100/100
theta:level 3–1	Chronos-2	-0.493 [-0.521, -0.466]	100/100
gain:level 3–1	ARX	-0.000 [-0.000, 0.000]	100/100
gain:level 3–1	FIR	-0.000 [-0.000, 0.000]	100/100
gain:level 3–1	MLP	-0.000 [-0.000, 0.000]	100/100
gain:level 3–1	RBF-SVR	-0.000 [-0.002, 0.001]	100/100
gain:level 3–1	TimesFM-3	0.002 [-0.000, 0.005]	100/100
gain:level 3–1	TiRex-2	0.000 [-0.002, 0.002]	100/100
gain:level 3–1	Chronos-2	0.001 [-0.001, 0.003]	100/100
mixing:level 3–1	ARX	-0.305 [-0.314, -0.296]	100/100
mixing:level 3–1	FIR	-0.305 [-0.314, -0.297]	100/100
mixing:level 3–1	MLP	-0.166 [-0.180, -0.152]	100/100
mixing:level 3–1	RBF-SVR	-0.073 [-0.086, -0.059]	100/100
mixing:level 3–1	TimesFM-3	-0.118 [-0.137, -0.099]	100/100
mixing:level 3–1	TiRex-2	-0.087 [-0.105, -0.068]	100/100
mixing:level 3–1	Chronos-2	-0.080 [-0.093, -0.065]	100/100
tank:level 3–1	ARX	-0.582 [-0.603, -0.563]	100/100
tank:level 3–1	FIR	-0.587 [-0.609, -0.568]	100/100
tank:level 3–1	MLP	-0.252 [-0.263, -0.240]	100/100
tank:level 3–1	RBF-SVR	-0.078 [-0.092, -0.063]	100/100
tank:level 3–1	TimesFM-3	-0.210 [-0.229, -0.190]	100/100
tank:level 3–1	TiRex-2	-0.150 [-0.165, -0.133]	100/100
tank:level 3–1	Chronos-2	-0.122 [-0.138, -0.103]	100/100
zeta:level 3–1	ARX	0.000 [-0.000, 0.000]	100/100
zeta:level 3–1	FIR	-0.000 [-0.000, -0.000]	100/100
zeta:level 3–1	MLP	-0.018 [-0.020, -0.016]	100/100
zeta:level 3–1	RBF-SVR	-0.016 [-0.025, -0.008]	100/100
zeta:level 3–1	TimesFM-3	-0.021 [-0.039, -0.004]	100/100
zeta:level 3–1	TiRex-2	0.065 [0.050, 0.081]	100/100
zeta:level 3–1	Chronos-2	0.202 [0.188, 0.216]	100/100
integrating:level 3–1	ARX	0.000 [-0.000, 0.000]	100/100
integrating:level 3–1	FIR	0.041 [-0.058, 0.155]	100/100
integrating:level 3–1	MLP	0.014 [-0.246, 0.281]	100/100
integrating:level 3–1	RBF-SVR	0.114 [-0.093, 0.318]	100/100
integrating:level 3–1	TimesFM-3	0.003 [-0.028, 0.033]	100/100
integrating:level 3–1	TiRex-2	-0.045 [-0.078, -0.011]	100/100
integrating:level 3–1	Chronos-2	-0.011 [-0.029, 0.008]	100/100
controller:aggressive-nominal	ARX	-0.000 [-0.000, -0.000]	100/100
controller:aggressive-nominal	FIR	-0.000 [-0.000, -0.000]	100/100
controller:aggressive-nominal	MLP	-0.051 [-0.068, -0.036]	100/100
controller:aggressive-nominal	RBF-SVR	0.114 [0.055, 0.164]	100/100
controller:aggressive-nominal	TimesFM-3	-0.406 [-0.468, -0.330]	100/100
controller:aggressive-nominal	TiRex-2	-0.506 [-0.609, -0.396]	100/100
controller:aggressive-nominal	Chronos-2	-0.308 [-0.373, -0.242]	100/100
colored:50–12.5%	ARX	-0.006 [-0.007, -0.005]	400/400
colored:50–12.5%	Chronos-2	0.072 [0.060, 0.085]	400/400
colored:50–12.5%	FIR	-0.007 [-0.008, -0.006]	400/400
colored:50–12.5%	MLP	-0.072 [-0.080, -0.064]	400/400
colored:50–12.5%	RBF-SVR	-0.046 [-0.053, -0.040]	400/400
colored:50–12.5%	TimesFM-3	0.021 [0.010, 0.033]	400/400
colored:50–12.5%	TiRex-2	0.020 [0.011, 0.028]	400/400
drift:50–12.5%	ARX	-0.004 [-0.005, -0.003]	400/400
drift:50–12.5%	Chronos-2	0.003 [-0.003, 0.008]	400/400
drift:50–12.5%	FIR	-0.015 [-0.018, -0.012]	400/400
drift:50–12.5%	MLP	-0.051 [-0.058, -0.044]	400/400
drift:50–12.5%	RBF-SVR	-0.082 [-0.096, -0.068]	400/400
drift:50–12.5%	TimesFM-3	-0.005 [-0.010, 0.001]	400/400
drift:50–12.5%	TiRex-2	0.008 [0.004, 0.013]	400/400
offset:50–12.5%	ARX	0.002 [0.000, 0.003]	400/400
offset:50–12.5%	Chronos-2	0.006 [-0.001, 0.013]	400/400
offset:50–12.5%	FIR	-0.020 [-0.023, -0.016]	400/400
offset:50–12.5%	MLP	-0.066 [-0.076, -0.057]	400/400

Continued from previous page

Contrast	Model	ΔMV skill [95% CI]	Groups L/R
offset:50–12.5%	RBF-SVR	-0.150 [-0.178, -0.126]	400/400
offset:50–12.5%	TimesFM-3	-0.013 [-0.020, -0.006]	400/400
offset:50–12.5%	TiRex-2	0.008 [0.003, 0.013]	400/400
scale:50–12.5%	ARX	-0.055 [-0.061, -0.050]	400/400
scale:50–12.5%	Chronos-2	0.001 [-0.006, 0.009]	400/400
scale:50–12.5%	FIR	-0.061 [-0.066, -0.055]	400/400
scale:50–12.5%	MLP	-0.081 [-0.090, -0.071]	400/400
scale:50–12.5%	RBF-SVR	-0.128 [-0.151, -0.106]	400/400
scale:50–12.5%	TimesFM-3	-0.018 [-0.024, -0.011]	400/400
scale:50–12.5%	TiRex-2	-0.001 [-0.007, 0.006]	400/400
soft_clip:50–12.5%	ARX	-0.048 [-0.053, -0.044]	400/400
soft_clip:50–12.5%	Chronos-2	-0.027 [-0.035, -0.019]	400/400
soft_clip:50–12.5%	FIR	-0.051 [-0.056, -0.045]	400/400
soft_clip:50–12.5%	MLP	-0.098 [-0.112, -0.087]	400/400
soft_clip:50–12.5%	RBF-SVR	-0.169 [-0.194, -0.142]	400/400
soft_clip:50–12.5%	TimesFM-3	-0.032 [-0.041, -0.024]	400/400
soft_clip:50–12.5%	TiRex-2	-0.023 [-0.030, -0.015]	400/400
load:0.25:measured-hidden	ARX	0.033 [0.029, 0.036]	300/300
load:0.25:measured-hidden	Chronos-2	0.040 [0.028, 0.052]	300/300
load:0.25:measured-hidden	FIR	0.054 [0.049, 0.060]	300/300
load:0.25:measured-hidden	MLP	0.235 [0.224, 0.248]	300/300
load:0.25:measured-hidden	RBF-SVR	0.242 [0.222, 0.266]	300/300
load:0.25:measured-hidden	TimesFM-3	0.007 [-0.002, 0.016]	300/300
load:0.25:measured-hidden	TiRex-2	-0.002 [-0.005, 0.000]	300/300
load:0.5:measured-hidden	ARX	0.067 [0.060, 0.074]	300/300
load:0.5:measured-hidden	Chronos-2	0.086 [0.072, 0.099]	300/300
load:0.5:measured-hidden	FIR	0.108 [0.098, 0.119]	300/300
load:0.5:measured-hidden	MLP	0.478 [0.455, 0.501]	300/300
load:0.5:measured-hidden	RBF-SVR	0.463 [0.429, 0.500]	300/300
load:0.5:measured-hidden	TimesFM-3	0.051 [0.039, 0.064]	300/300
load:0.5:measured-hidden	TiRex-2	-0.012 [-0.020, -0.004]	300/300
load:1.0:measured-hidden	ARX	0.138 [0.122, 0.154]	300/300
load:1.0:measured-hidden	Chronos-2	0.196 [0.178, 0.215]	300/300
load:1.0:measured-hidden	FIR	0.212 [0.190, 0.233]	300/300
load:1.0:measured-hidden	MLP	0.971 [0.919, 1.028]	300/300
load:1.0:measured-hidden	RBF-SVR	0.798 [0.749, 0.846]	300/300
load:1.0:measured-hidden	TimesFM-3	0.118 [0.100, 0.136]	300/300
load:1.0:measured-hidden	TiRex-2	-0.142 [-0.166, -0.120]	300/300

E CONTROL OUTCOMES AND STRESS CONTRASTS

Table 18: **Core control summary.** Error, nTV, and cost use the same 2,166 common complete tasks. Termination and fallback counts use all 2,228 core tasks; the two outcomes can overlap. References use the distinct information and feedback protocols described above.

Model	nIAE ↓	95% interval	nTV ↓	J ↓	Term.	Fallback
ARX	0.2350	[0.2292, 0.2409]	0.2253	0.2462	0	4
FIR	0.3348	[0.3241, 0.3461]	0.2142	0.3455	1	5
MLP	0.3041	[0.2958, 0.3128]	0.2055	0.3144	1	0
RBF-SVR	0.7869	[0.7454, 0.8302]	0.2473	0.7992	14	0
TimesFM-3	0.3659	[0.3553, 0.3768]	0.7483	0.4033	25	60
TiRex-2	0.3453	[0.3348, 0.3560]	0.5732	0.3739	6	27
Chronos-2	0.5623	[0.5450, 0.5804]	1.1321	0.6189	41	19
True-model MPC	0.2278	[0.2223, 0.2334]	0.2226	0.2389	0	0
PI	0.1468	[0.1422, 0.1516]	0.0282	0.1482	0	12

Failure concentration and common support. Across nine configurations, the 113 test-run terminations comprise 112 outflow cases and one RBF-SVR integrating case. In core tracking, 88 configuration-level terminations correspond to 62 unique tasks excluded from the common complete

Table 19: **Family-specific core nIAE.** Means condition on common completion. Outflow includes only 212/273 tasks and must be read with its termination counts. Each family has its own support and aggregation.

Model	FOPDT	Mixing	Outflow	SOPDT	Integrating
ARX	0.2508	0.2562	0.2533	0.2432	0.1715
FIR	0.2508	0.2640	0.2651	0.2432	0.6508
MLP	0.2523	0.2589	0.2271	0.2401	0.5422
RBF-SVR	0.8835	0.8926	0.8461	0.6943	0.6179
TimesFM-3	0.4010	0.4983	0.3556	0.3399	0.2347
TiRex-2	0.3604	0.3977	0.3003	0.4020	0.2658
Chronos-2	0.5625	0.8942	0.4933	0.5909	0.2704
True-model MPC	0.2508	0.2706	0.2028	0.2432	0.1715
PI	0.1350	0.1380	0.1258	0.1939	0.1413

Table 20: **Outcomes across all test modules.** Denominator: 5,594 tasks per configuration. Counts on shared tasks are not independent events or independent physical systems.

Model	Terminations / 5,594	Raw rate	Fallback episodes
ARX	0	0.000%	37
FIR	1	0.018%	5
MLP	1	0.018%	4
RBF-SVR	14	0.250%	0
TimesFM-3	32	0.572%	227
TiRex-2	10	0.179%	92
Chronos-2	55	0.983%	78
True-model MPC	0	0.000%	0
PI	0	0.000%	12

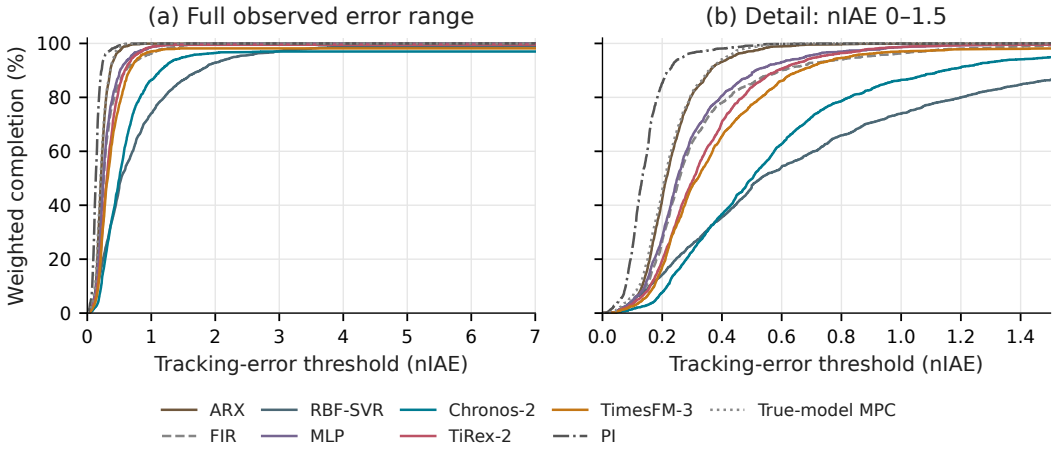

Figure 8: **Full-denominator core attainment.** The original task weights are retained; termination fails every finite error threshold. Curves show the weighted fraction of all 2,228 core tasks that complete with nIAE at or below the horizontal-axis value. (a) Full observed error range. (b) Detail of thresholds 0–1.5. No area-under-curve score is used. Plateaus retain termination losses and do not imply equal family-specific reliability.

set. Of 273 outflow tasks, Chronos-2 terminates on 41, TimesFM-3 on 25, RBF-SVR on 13, TiRex-2 on six, and FIR and MLP on one each. Most occur at the strongest tested outflow nonlinearity ($\gamma = 0.1$). These counts describe the fixed inventory; different configurations can terminate on the same task. Keeping only whole groups with all matched levels complete leaves 871 groups and 2,053 tasks and preserves the learned-model mean order, but does not remove the conditional nature of error summaries.

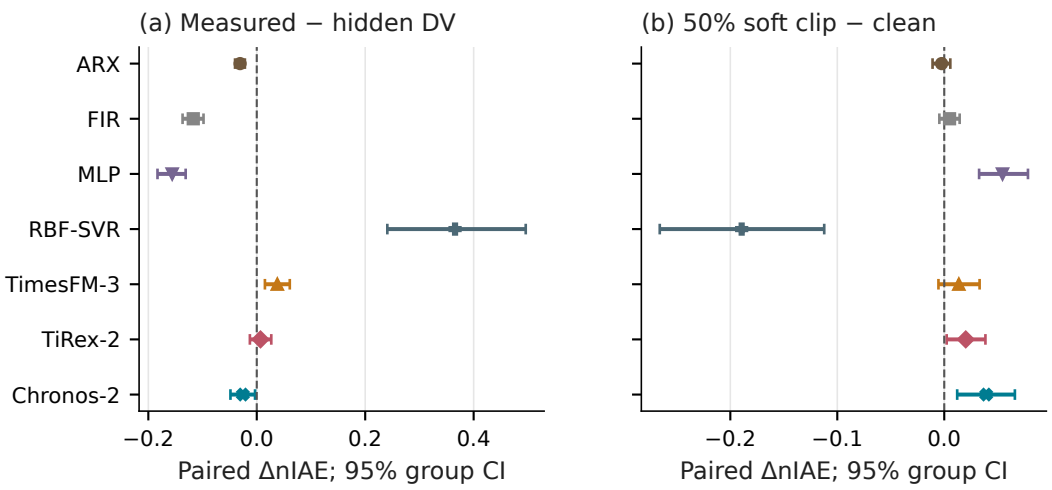

Figure 9: **Paired control stress contrasts.** Negative values favor the first condition. (a) Event-period measured-minus-hidden DV on 222/224 common complete groups. (b) Historical soft clipping at 50% coverage minus clean history on 296/297 common groups. Bars are exploratory 95% paired-group intervals. Both compare corresponding information interfaces and fitted models; neither isolates an internal modeling mechanism.

E.1 What do the stress-module averages contain?

Table 21 shows absolute nIAE by module. Each includes its own reference conditions and common-completion set. The columns therefore cannot be subtracted to estimate the effect of a stressor. Tables 22–24 expose those conditions, and Table 25 supplies the actual paired changes. Pattern-history tasks preserve their observed terminal states; unlike the simulation pattern construction, they are not a pure terminal-state-matched intervention on history.

Table 21: **Absolute control error by module.** All nine configurations use the displayed common-completion denominator in each column.

Model	Nominal	Pattern history	Corruption	Load
ARX	0.2067	0.3676	0.2509	0.1544
FIR	0.2067	0.3522	0.2572	0.1880
MLP	0.2090	0.3628	0.2691	0.2044
RBF-SVR	0.8497	0.6430	0.7643	0.6992
TimesFM-3	0.3504	0.7433	0.3892	0.3537
TiRex-2	0.2852	0.7642	0.3642	0.2524
Chronos-2	0.4998	1.1430	0.6200	0.3533
True-model MPC	0.2067	0.3547	0.2320	0.1374
PI	0.1218	0.1321	0.1499	0.0901
Common / planned	94/94	1173/1188	1182/1188	894/896

Table 22: **Control after different history patterns.** Values are nIAE. Each pattern has its own common-completion set; paired contrasts use the intersection with step history.

Model	Step	Ramp	Pulse	Sine
ARX	0.3617	0.3796	0.3502	0.3788
FIR	0.3494	0.3494	0.3516	0.3556
MLP	0.3471	0.3769	0.3518	0.3731
RBF-SVR	0.6757	0.7467	0.4916	0.6566
TimesFM-3	0.8611	0.5697	0.7498	0.8018
TiRex-2	0.6618	0.7951	0.7486	0.8438
Chronos-2	1.2386	1.0961	1.1120	1.1232
True-model MPC	0.3542	0.3547	0.3550	0.3554
PI	0.1324	0.1320	0.1327	0.1316
Common / planned	289/297	297/297	294/297	293/297

Table 23: **Control after historical soft clipping.** Coverage refers to the corrupted historical segment. Executed future feedback is clean.

Model	Clean	12.5%	25%	50%
ARX	0.2519	0.2518	0.2515	0.2494
FIR	0.2563	0.2540	0.2581	0.2612
MLP	0.2470	0.2557	0.2737	0.3012
RBF-SVR	0.8418	0.8101	0.7637	0.6521
TimesFM-3	0.3766	0.3963	0.3956	0.3900
TiRex-2	0.3522	0.3566	0.3778	0.3717
Chronos-2	0.6063	0.6091	0.6223	0.6457
True-model MPC	0.2318	0.2328	0.2323	0.2320
PI	0.1456	0.1484	0.1507	0.1556
Common / planned	296/297	293/297	296/297	297/297

Table 24: **Load-event and information-view controls.** Entries are nIAE. Event/no-event and measured/hidden comparisons require their paired common support.

Model	Hidden/no event	Measured/no event	Hidden/event	Measured/event
ARX	0.1512	0.1338	0.1811	0.1511
FIR	0.1987	0.1363	0.2679	0.1503
MLP	0.2400	0.1320	0.3012	0.1456
RBF-SVR	0.5157	0.8273	0.5482	0.9083
TimesFM-3	0.3248	0.3488	0.3525	0.3897
TiRex-2	0.2121	0.2540	0.2679	0.2755
Chronos-2	0.3513	0.3113	0.3882	0.3624
True-model MPC	0.1257	0.1257	0.1496	0.1490
PI	0.0823	0.0823	0.0981	0.0979
Common / planned	224/224	224/224	222/224	224/224

Distinguishing event error from event suppression. Measured-minus-hidden event-period nIAE is -0.0308 for ARX, -0.1171 for FIR, -0.1560 for MLP, and -0.0257 for Chronos-2; each corresponding interval excludes zero. The contrasts are $+0.3657$ for RBF-SVR, $+0.0379$ for TimesFM-3, and $+0.0069$ for TiRex-2, the last with interval $[-0.0127, 0.0268]$. The difference of event-minus-no-event changes,

$$\Delta_{\mathrm{DV}} = (E_{m,\mathrm{event}} - E_{m,\mathrm{none}}) - (E_{h,\mathrm{event}} - E_{h,\mathrm{none}}),$$

answers a narrower question about the added event. On the same 222 common groups, TiRex-2 has $\Delta_{\mathrm{DV}} = -0.0348$ ($[-0.0613, -0.0090]$), whereas Chronos-2 has $+0.0162$ ($[-0.0189, 0.0504]$). PI does not consume DV, and the true-model reference has current DV in both views, so their zero cross-view differences are expected interface controls.

Table 25: **Complete paired control stress contrasts.** Corruption is relative to clean history; pattern is relative to step history; load labels state the subtraction. Negative values favor the first named condition. These are exploratory group intervals.

Contrast	Model	ΔnIAE [95% CI]	Pairs / total
Clip minus clean: 0.125	ARX	-0.001 [-0.006, 0.004]	293/297
Clip minus clean: 0.125	FIR	-0.003 [-0.009, 0.003]	293/297
Clip minus clean: 0.125	RBF-SVR	-0.038 [-0.100, 0.023]	293/297
Clip minus clean: 0.125	MLP	0.008 [-0.003, 0.019]	293/297
Clip minus clean: 0.125	Chronos-2	0.001 [-0.023, 0.024]	293/297
Clip minus clean: 0.125	TiRex-2	0.004 [-0.009, 0.017]	293/297
Clip minus clean: 0.125	TimesFM-3	0.019 [-0.001, 0.039]	293/297
Clip minus clean: 0.125	PI	0.002 [0.000, 0.004]	293/297
Clip minus clean: 0.125	True-model MPC	0.000 [0.000, 0.000]	293/297
Clip minus clean: 0.25	ARX	-0.001 [-0.008, 0.007]	295/297
Clip minus clean: 0.25	FIR	0.002 [-0.006, 0.009]	295/297
Clip minus clean: 0.25	RBF-SVR	-0.079 [-0.148, -0.007]	295/297
Clip minus clean: 0.25	MLP	0.028 [0.013, 0.044]	295/297

Continued from previous page

Contrast	Model	ΔnIAE [95% CI]	Pairs / total
Clip minus clean: 0.25	Chronos-2	0.017 [-0.011, 0.044]	295/297
Clip minus clean: 0.25	TiRex-2	0.026 [0.011, 0.041]	295/297
Clip minus clean: 0.25	TimesFM-3	0.018 [-0.000, 0.037]	295/297
Clip minus clean: 0.25	PI	0.005 [0.002, 0.008]	295/297
Clip minus clean: 0.25	True-model MPC	0.000 [0.000, 0.000]	295/297
Clip minus clean: 0.5	ARX	-0.002 [-0.011, 0.006]	296/297
Clip minus clean: 0.5	FIR	0.005 [-0.005, 0.014]	296/297
Clip minus clean: 0.5	RBF-SVR	-0.189 [-0.266, -0.112]	296/297
Clip minus clean: 0.5	MLP	0.054 [0.033, 0.078]	296/297
Clip minus clean: 0.5	Chronos-2	0.039 [0.012, 0.066]	296/297
Clip minus clean: 0.5	TiRex-2	0.020 [0.002, 0.038]	296/297
Clip minus clean: 0.5	TimesFM-3	0.014 [-0.005, 0.033]	296/297
Clip minus clean: 0.5	PI	0.010 [0.006, 0.015]	296/297
Clip minus clean: 0.5	True-model MPC	0.000 [0.000, 0.000]	296/297
Pattern minus step: pulse	ARX	-0.011 [-0.021, -0.003]	288/297
Pattern minus step: pulse	FIR	0.003 [-0.003, 0.009]	288/297
Pattern minus step: pulse	RBF-SVR	-0.183 [-0.241, -0.124]	288/297
Pattern minus step: pulse	MLP	0.005 [-0.005, 0.015]	288/297
Pattern minus step: pulse	Chronos-2	-0.137 [-0.193, -0.080]	288/297
Pattern minus step: pulse	TiRex-2	0.083 [0.042, 0.122]	288/297
Pattern minus step: pulse	TimesFM-3	-0.113 [-0.157, -0.070]	288/297
Pattern minus step: pulse	PI	0.000 [-0.001, 0.001]	288/297
Pattern minus step: pulse	True-model MPC	-0.000 [-0.000, 0.000]	288/297
Pattern minus step: ramp	ARX	0.019 [0.004, 0.035]	289/297
Pattern minus step: ramp	FIR	-0.001 [-0.009, 0.006]	289/297
Pattern minus step: ramp	RBF-SVR	0.070 [-0.008, 0.148]	289/297
Pattern minus step: ramp	MLP	0.029 [0.016, 0.042]	289/297
Pattern minus step: ramp	Chronos-2	-0.146 [-0.201, -0.093]	289/297
Pattern minus step: ramp	TiRex-2	0.134 [0.091, 0.178]	289/297
Pattern minus step: ramp	TimesFM-3	-0.288 [-0.333, -0.243]	289/297
Pattern minus step: ramp	PI	-0.000 [-0.001, 0.001]	289/297
Pattern minus step: ramp	True-model MPC	-0.000 [-0.000, 0.000]	289/297
Pattern minus step: sine	ARX	0.018 [0.002, 0.035]	287/297
Pattern minus step: sine	FIR	0.005 [-0.003, 0.013]	287/297
Pattern minus step: sine	RBF-SVR	-0.015 [-0.085, 0.055]	287/297
Pattern minus step: sine	MLP	0.025 [0.012, 0.038]	287/297
Pattern minus step: sine	Chronos-2	-0.124 [-0.179, -0.069]	287/297
Pattern minus step: sine	TiRex-2	0.178 [0.132, 0.224]	287/297
Pattern minus step: sine	TimesFM-3	-0.062 [-0.104, -0.019]	287/297
Pattern minus step: sine	PI	-0.001 [-0.002, 0.000]	287/297
Pattern minus step: sine	True-model MPC	0.000 [-0.000, 0.000]	287/297
Hidden: event minus none	ARX	0.030 [0.016, 0.043]	222/224
Hidden: event minus none	FIR	0.069 [0.049, 0.087]	222/224
Hidden: event minus none	RBF-SVR	0.031 [0.008, 0.052]	222/224
Hidden: event minus none	MLP	0.061 [0.040, 0.081]	222/224
Hidden: event minus none	Chronos-2	0.037 [0.013, 0.061]	222/224
Hidden: event minus none	TiRex-2	0.056 [0.035, 0.077]	222/224
Hidden: event minus none	TimesFM-3	0.027 [0.005, 0.048]	222/224
Hidden: event minus none	PI	0.016 [0.009, 0.022]	222/224
Hidden: event minus none	True-model MPC	0.024 [0.013, 0.034]	222/224
Measured: event minus none	ARX	0.017 [0.005, 0.029]	224/224
Measured: event minus none	FIR	0.014 [0.002, 0.026]	224/224
Measured: event minus none	RBF-SVR	0.081 [-0.014, 0.175]	224/224
Measured: event minus none	MLP	0.014 [0.003, 0.024]	224/224
Measured: event minus none	Chronos-2	0.051 [0.023, 0.078]	224/224
Measured: event minus none	TiRex-2	0.022 [0.000, 0.043]	224/224
Measured: event minus none	TimesFM-3	0.041 [0.018, 0.065]	224/224
Measured: event minus none	PI	0.016 [0.009, 0.022]	224/224
Measured: event minus none	True-model MPC	0.023 [0.013, 0.034]	224/224
No event: measured minus hidden	ARX	-0.017 [-0.026, -0.009]	224/224
No event: measured minus hidden	FIR	-0.062 [-0.083, -0.046]	224/224
No event: measured minus hidden	RBF-SVR	0.312 [0.191, 0.438]	224/224

Continued from previous page

Contrast	Model	ΔnIAE [95% CI]	Pairs / total
No event: measured minus hidden	MLP	-0.108 [-0.130, -0.088]	224/224
No event: measured minus hidden	Chronos-2	-0.040 [-0.067, -0.012]	224/224
No event: measured minus hidden	TiRex-2	0.042 [0.022, 0.063]	224/224
No event: measured minus hidden	TimesFM-3	0.024 [0.002, 0.046]	224/224
No event: measured minus hidden	PI	0.000 [0.000, 0.000]	224/224
No event: measured minus hidden	True-model MPC	0.000 [0.000, 0.000]	224/224
Event: measured minus hidden	ARX	-0.031 [-0.040, -0.022]	222/224
Event: measured minus hidden	FIR	-0.117 [-0.137, -0.098]	222/224
Event: measured minus hidden	RBF-SVR	0.366 [0.241, 0.496]	222/224
Event: measured minus hidden	MLP	-0.156 [-0.183, -0.131]	222/224
Event: measured minus hidden	Chronos-2	-0.026 [-0.049, -0.003]	222/224
Event: measured minus hidden	TiRex-2	0.007 [-0.013, 0.027]	222/224
Event: measured minus hidden	TimesFM-3	0.038 [0.015, 0.061]	222/224
Event: measured minus hidden	PI	0.000 [0.000, 0.000]	222/224
Event: measured minus hidden	True-model MPC	0.000 [0.000, 0.000]	222/224

F PAIRED SIMULATION–CONTROL ANALYSIS

Population and weights. The nominal, gain, time-constant, and delay sources each contribute 94 test groups, giving 376 groups and 940 tasks. Each task compares the same seven models on a common in-range probe set; the 2,994 probes per model are not independent physical systems. Within-task model pairs, within-group levels, within-source groups, and the four sources are weighted equally at their respective levels. Bootstrap replicates resample groups within sources 10,000 times (seed 20260925), carrying every model and condition together. Control, both skills, and all four response diagnostics have complete support: no terminated controls, missing response features, or undetectable tail responses occur in this selected population. This does not imply the same coverage outside clean FOPDT.

Orderings and ties. Within a task, higher simulation skill corresponds to lower mean prediction error because the reference magnitude is shared across models. We compare that skill ordering with lower executed nIAE. The tie tolerance $\varepsilon_{\mathrm{tie}} = 10^{-10}$ is applied to differences in skill and nIAE, not to differences in raw prediction error. Aligned, reversed, simulation-only tie, control-only tie, and double tie categories exhaust the comparison weight. The earlier skill-based agreement measure gives any tie half credit. Under PV-RMSE, the raw reversal count is 4,608 pairs, involving 906 tasks and 369 groups; the weighted reversal share is 23.185%. The 69.586% agreement for both PV and MV skill is not a copied score: opposing source-level differences cancel under the fixed source weights. Their difference interval is $[-0.199, 0.198]$ percentage points.

Table 26: **Ordering reversal rates with exploratory 95% intervals.** All-comparison rates preserve ties in the denominator; the right column conditions on neither side being tied. Four variants use the same tasks and group weights.

Metric	All comparisons (%)	Excluding ties (%)
PV / RMSE	23.18 [22.21, 24.17]	27.10 [26.02, 28.20]
PV / MAE	23.07 [22.11, 24.05]	26.98 [25.91, 28.07]
MV / RMSE	23.18 [22.21, 24.19]	27.10 [26.04, 28.21]
MV / MAE	23.12 [22.13, 24.12]	27.03 [25.95, 28.13]

Control differences within reversed pairs. For a reversed pair, define $\Delta\mathrm{nIAE}$ as the nIAE of the predictor with lower mean PV-RMSE minus that of the other predictor. We retain the original pair weights and normalize them within the reversed subset. Across 4,608 reversed pairs, its weighted mean is 0.395, with an exploratory 95% group-bootstrap interval of $[0.362, 0.429]$, and its weighted median is 0.129. Bootstrap draws resample groups within the four sources and recompute both the conditional numerator and denominator. This is a descriptive difference conditional on reversal, not the regret of choosing the best of seven predictors or an effect of improving physical fidelity.

Table 27 separates all 21 model pairs. RBF-SVR paired with the three TSFMs accounts for 40.17% of the reversal weight and 83.06% of the weighted difference numerator; hence the overall mean is not

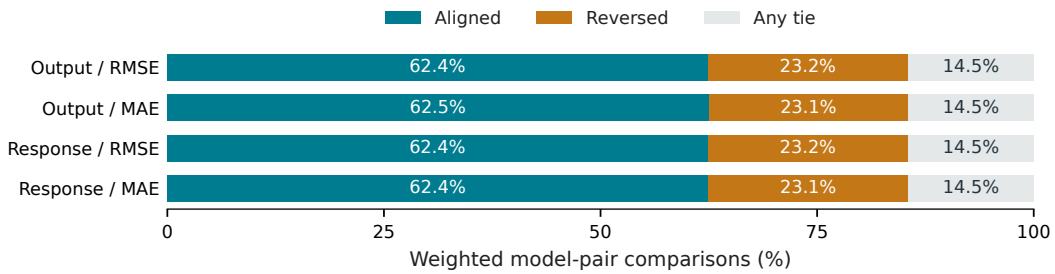

Figure 10: **Rankings of recorded configurations can differ.** Seven-model comparisons on 940 clean FOPDT tasks; TSFM runtime identities have the limits in Appendix C.2. "Aligned" means lower simulation error coincides with lower control nIAE; "reversed" means higher nIAE. Ties remain in the denominator.

typical of every pair. Table 28 retains the same tasks and source/group weights while renormalizing within each selected set of model pairs. Reversals also occur among the local models with bound fitted artifacts and after excluding RBF-SVR. These point estimates do not establish historical TSFM runtime identity or a causal explanation; the limitations in Appendix C.2 still apply.

As an exploratory magnitude check, requiring a reversed pair to have $\Delta\text{nIAE} > 0.01$, 0.05, or 0.10 leaves 21.69%, 16.95%, or 13.15% of all weighted comparisons, respectively. The denominator and numerical tie rule remain unchanged. These thresholds describe control-gap sensitivity, not a universal practical-significance criterion; no threshold-specific intervals have been computed.

Table 27: **Reversals by model pair.** Each pair is evaluated on 940 tasks. Rates retain ties in the denominator and use the original task weights. Mean and median ΔnIAE condition on reversal within that pair; a dash denotes no reversed tasks. These are descriptive point estimates.

Model pair	Reversed tasks	Rate (%)	Mean ΔnIAE	Median ΔnIAE
ARX / FIR	0	0.00	–	–
ARX / RBF-SVR	131	13.03	0.084	0.068
ARX / MLP	71	7.54	0.053	0.035
ARX / Chronos-2	62	6.03	0.071	0.059
ARX / TiRex-2	218	23.58	0.066	0.050
ARX / TimesFM-3	149	15.51	0.058	0.034
FIR / RBF-SVR	131	13.03	0.084	0.068
FIR / MLP	71	7.54	0.053	0.035
FIR / Chronos-2	62	6.03	0.071	0.059
FIR / TiRex-2	218	23.58	0.066	0.050
FIR / TimesFM-3	149	15.51	0.058	0.034
RBF-SVR / MLP	191	20.12	0.404	0.109
RBF-SVR / Chronos-2	545	57.54	0.813	0.649
RBF-SVR / TiRex-2	659	71.37	0.820	0.660
RBF-SVR / TimesFM-3	624	66.67	0.816	0.638
MLP / Chronos-2	63	6.29	0.070	0.057
MLP / TiRex-2	227	24.73	0.069	0.052
MLP / TimesFM-3	157	16.22	0.057	0.033
Chronos-2 / TiRex-2	190	19.15	0.135	0.100
Chronos-2 / TimesFM-3	224	24.11	0.107	0.073
TiRex-2 / TimesFM-3	466	49.29	0.151	0.125

Table 28: **Reversal rates in model subsets.** The same 940 tasks and group weights are retained, with model pairs weighted equally within each subset. Ties remain in the denominator. Subset intervals have not been computed.

Model subset	Pairs per task	Reversal (%)
All seven	21	23.18
Local models only	6	10.21
TSFMs only	3	30.85
Without RBF-SVR	15	16.34

Response direction over time. On the same clean-FOPDT probes, let $A_\star = K\Delta u$ be the known physical step effect and $s_\star = \mathrm{sign}(A_\star)$. With samples indexed $k = 1, \ldots, H$, the first nonzero physical response occurs at $k_0 = \theta + 1$. Write $\mathcal{I} = \{k_0, \ldots, H\}$. The reversed-response duration and mean reversed-response magnitude are

$$D_{\mathrm{dir}} = \frac{1}{|\mathcal{I}|} \sum_{k \in \mathcal{I}} \mathbf{1}\{s_\star \hat{d}_k < -\varepsilon_{\mathrm{dir}} |A_\star|\}, \qquad M_{\mathrm{rev}} = \frac{1}{|\mathcal{I}|} \sum_{k \in \mathcal{I}} \frac{\max(0, -s_\star \hat{d}_k)}{|A_\star|}.$$

We use the relative tolerance $\varepsilon_{\mathrm{dir}} = 10^{-6}$ and retain sensitivity checks at 0.01 and 0.05. Figure 3b reports $100 D_{\mathrm{dir}}$. This checks the side of the hold response on which the predicted action effect lies, not the sign of its time derivative. The true parameters determine offline scores only; models do not receive them. Metrics are averaged over eligible probes within each task, then with the source and group weights above.

Premature and weak responses. Before k_0, the true incremental response is zero, so we report $L_{\mathrm{pre}} = \mathrm{RMS}_{k<k_0}(\hat{d}_k)/|A_\star|$ separately. Define $\mathcal{I}_{\mathrm{active}} = \{k \in \mathcal{I} : |d_k| \geq \rho_{\mathrm{active}} |A_\star|\}$. The weak-response fraction is $W = |\mathcal{I}_{\mathrm{active}}|^{-1} \sum_{k \in \mathcal{I}_{\mathrm{active}}} \mathbf{1}\{|\hat{d}_k| < \rho_{\mathrm{weak}} |d_k|\}$, with $\rho_{\mathrm{active}} = \rho_{\mathrm{weak}} = 0.05$. A zero prediction has no reversed response but $W = 1$; small negative predictions can be both weak and reversed, so these diagnostics are not mutually exclusive categories.

Table 29: **Response direction throughout the physical response window.** Each model has 2,994 probes on 940 clean-FOPDT tasks. Any counts probes with $D_{\mathrm{dir}} > 0$; Strong uses the 5% amplitude threshold. Counts are unweighted. The remaining columns are weighted means, scaled by 100: reversed duration, reversed magnitude, premature-response RMS, and weak-response fraction. These are descriptive point estimates; small values can round to zero.

Model	Any	Strong	$100 D_{\mathrm{dir}}$	$100 M_{\mathrm{rev}}$	$100 L_{\mathrm{pre}}$	$100 W$
ARX	0	0	0.0000	0.0000	0.0000	0.0000
FIR	0	0	0.0000	0.0000	0.0000	0.0000
MLP	0	0	0.0000	0.0000	0.5439	0.0000
RBF-SVR	2	0	0.0007	0.0000	0.8243	0.0000
TimesFM-3	121	48	0.0702	0.0044	5.4209	0.0374
TiRex-2	89	47	0.3181	0.0676	5.9108	0.1515
Chronos-2	265	137	2.4317	0.7536	7.1552	1.2399

Timing and amplitude. For an H-step incremental response z, define tail amplitude $A(z) = L_{\mathrm{tail}}^{-1} \sum_{k=H-L_{\mathrm{tail}}+1}^{H} z_k$. For $|A(z)| \geq \varepsilon_A$, let $t_p(z) = \min\{k : z_k / A(z) \geq p\}$. Onset error compares $t_{p_{\mathrm{on}}}$; rise-interval error compares $t_{p_{\mathrm{rise}}} - t_{p_{\mathrm{on}}}$; amplitude error is $|A(\hat{d}) - A(d)|$. We use $H = 96$, tail-window length $L_{\mathrm{tail}} = 24$, thresholds $p_{\mathrm{on}} = 0.05$ and $p_{\mathrm{rise}} = 0.632$, and minimum detectable amplitude $\varepsilon_A = 10^{-6}$ in PV units. Times are in steps. Each response uses its own tail amplitude to define crossings. These finite-window features do not estimate asymptotic physical parameters. Crossing times are undefined when the amplitude is undetectable or the threshold is never reached.

Table 30: **Timing and amplitude uncertainty.** Same 940 clean-FOPDT tasks and 2,994 probes per model as Figure 3b. All three diagnostics have complete support; intervals resample 376 groups. Near-zero values can round to zero.

Model	Onset (steps)	Rise (steps)	Amplitude (PV)
ARX	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]
FIR	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]	0.00 [0.00, 0.00]
MLP	0.12 [0.08, 0.17]	0.36 [0.31, 0.41]	0.01 [0.01, 0.01]
RBF-SVR	0.13 [0.07, 0.21]	0.41 [0.34, 0.50]	0.08 [0.08, 0.09]
TimesFM-3	2.59 [2.38, 2.79]	3.38 [3.16, 3.61]	0.31 [0.29, 0.33]
TiRex-2	3.25 [3.04, 3.47]	4.44 [4.18, 4.72]	0.52 [0.49, 0.55]
Chronos-2	5.75 [5.48, 6.03]	12.08 [11.43, 12.75]	1.10 [1.04, 1.16]

Probe and execution distributions. The response diagnostics use in-range steps, whereas executed MPC actions can leave the initial historical input range. For example, the within-range RBF-SVR response diagnostics can be accurate despite poor executed control; 89.37% of its execution steps

lie outside the initial historical MV range in this paired subset. This is an observed distribution difference, not proof of the cause of the control error. No post-control range filter is used to select favorable trajectories.

F.1 What does a prediction–control reversal look like?

Figure 11 shows prediction and execution trajectories for ordering reversals; Figure 6 presents its nominal-source row in the main text. Within each of the four FOPDT sources, cases are selected from model pairs that reverse under both RMSE and MAE. The displayed pair is closest to the source-weighted median control penalty, with deterministic task/model tie-breaking; the first eligible in-range probe is then shown. Thus selection is based on an explicit rule rather than visual appeal. The pair ordering uses mean error over all eligible probes, not necessarily the single plotted probe.

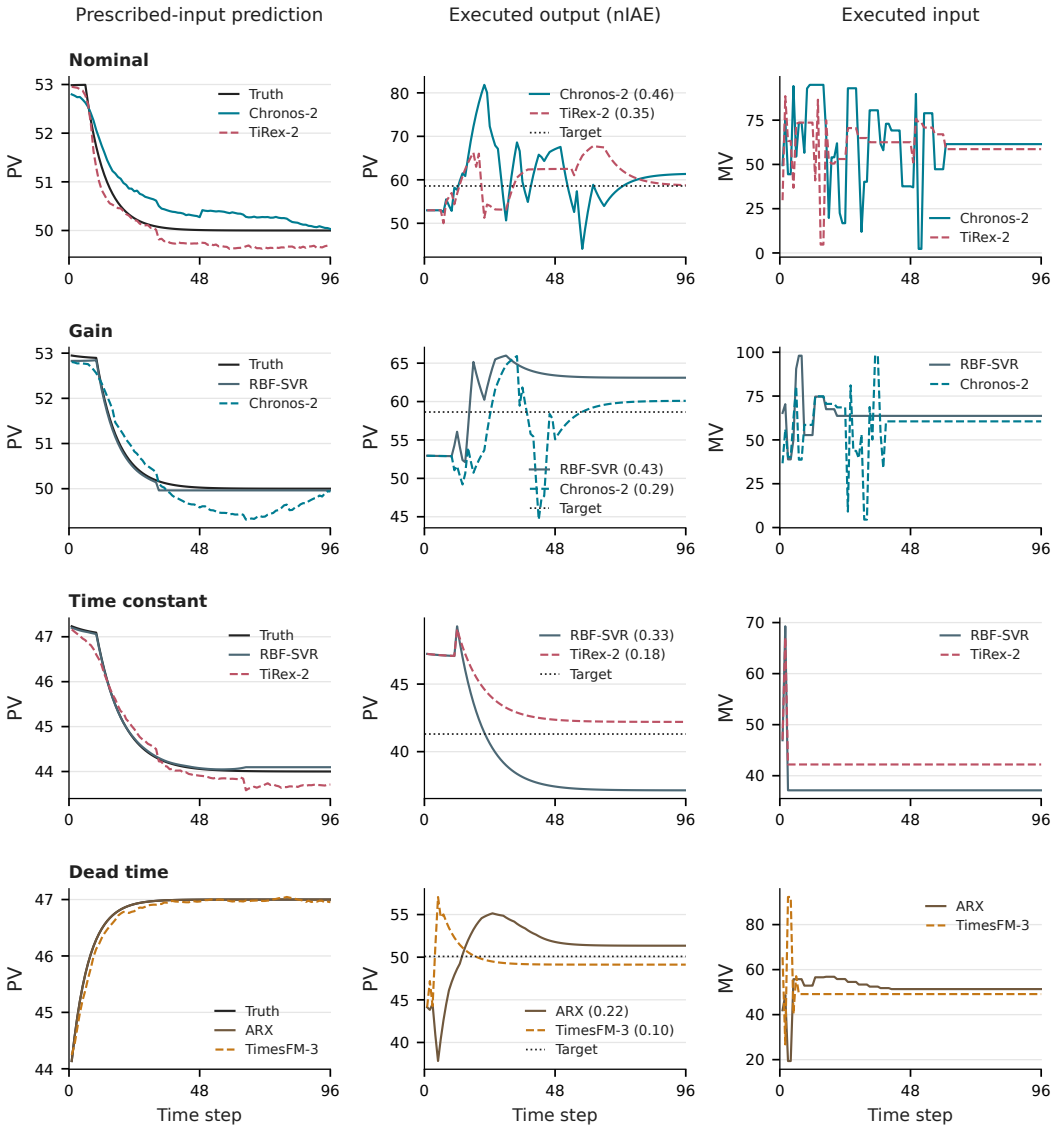

Figure 11: **Examples of prediction–control ordering reversal.** Each row compares two models on one task. The solid model curve has lower mean prescribed-input RMSE and MAE but higher executed nIAE than the dashed model curve. Left: one prescribed-input query and its truth. Middle/right: separately executed controller outputs and inputs. The input plans differ between prediction probes and execution. Legends in the middle column report nIAE. Cases illustrate behavior rather than its prevalence or cause; task/query identities and curves are archived with the figure.

G SOFTWARE WORKFLOW AND REUSE

The anonymous repository provides a public reference workflow for paired trajectory generation, predictor integration, executed control, and inspection (Figure 12). It uses the `DatasetSuite` format. The paper experiments instead use frozen `unified-public.v1/v2` data and a separate research runner. These workflows share the prediction-to-control design but do not have identical input contracts, controller settings, or aggregation. The reference CLI and explorer do not directly load the paper archives.

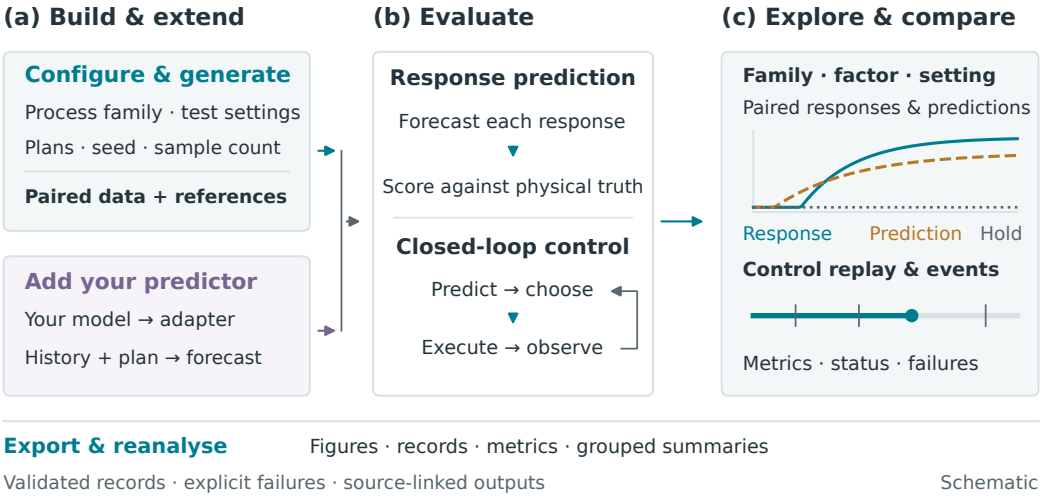

Figure 12: **Public reference workflow.** (a) Data and model adapters enter (b) response prediction and executed control. (c) The explorer supports condition filtering, response comparison, and recorded control replay. Exports retain scores, failures, and source identities; saved metrics support reaggregation without model reruns. The explorer uses reference-workflow records, not the frozen paper archives. Curves and panels are schematic.

Generate and evaluate reference data. A recipe specifies process conditions, seeds, and accepted examples per condition. The generator saves paired branches, histories, input plans, identities, and source hashes. The `validate` command checks the data contract and references. An external adapter exposes `model_descriptors()`, fits from the permitted history, and returns forecasts with query identities, horizons, information modes, and statuses. The history-only example at `examples/models/mean_forecaster.py` does not condition on planned actions and is therefore unsuitable for MPC. An action-conditioned adapter must declare and implement planned-MV conditioning and batched candidate rollout.

Execute new reference control runs. First generate and validate a suite as described in `documentation/usage.md`. Set the external descriptor name under `models` in a run YAML and configure `control.plugin: cem_mpc`; `config/run/quickstart_control.yaml` supplies a small CPU configuration. With `SUITE` denoting the generated manifest and `RUN_CONFIG` the resulting YAML, run:

```
python -m src run --dataset-suite-manifest "$SUITE" \
  --config "$RUN_CONFIG" --plugin adapter.py \
  --artifact-root artifacts/new-control
```

This starts new closed-loop execution. The adapter fits before execution and is held fixed within the episode. An integration check loaded the existing ARX implementation under an external descriptor and completed a new control episode using this route; it checks the interface, not the paper's experimental results. The small reference configuration is not the paper's 96-step, 32-by-2, four-step-replanning protocol.

Inspect, export, and recompute. The returned comparison directory supports `browse`, `report`, and `export`. Exported `control_traces.jsonl` contains recorded trajectories, `metrics.`

`json` contains scores and statuses, and `failures.json` retains failure records. Browsing replays saved control; it does not start a new controller. The `summarize` command reaggregates exported metrics without model execution. Inspect command status and failure records as well as output existence. The usage guide documents the adapter contract, installation, checkpoints, and commands.

Paper-protocol extension. The research entry points are `experiment/process_disturbance_models.py` for predictions, `experiment/score_formal_evaluation.py` for scoring, and `experiment/control_formal.py` for control. The control runner requires frozen task data, fitted-model bindings, and development registrations; its model list is explicit. A public `-plugin` adapter is not automatically available to this runner: adding a predictor requires a research-runner integration and preservation of the paper's PV/MV/DV interface and frozen episode fits.

Reconstructing the paper inventory. The accompanying source distribution includes the original 4,400 physical-history specifications in `config/paper/batch1_registration.json`. The first generator's `-registration` option replays these parameter values instead of resampling them; this avoids last-bit parameter differences and changed instance identities across random-number-library versions. The formal-generation sequence in `documentation/usage.md` then builds the input, corruption, load, and four-pattern modules and assembles the primary inventory. `config/paper/inventory.json` records the expected task-identity digest, 70 development groups, and 20 pattern-repair cases. After assembly, run:

```
python -m experiment.paper_release "$DATA" \
  --development-output "$DEVELOPMENT"
```

This verifies task identities and protocol metadata and rebuilds the two development-membership records consumed by formal control. Their existing deterministic selection rules are reused; rebuilding membership does not rerun the historical development experiments. The guide distinguishes control registration from execution and gives the prediction, control, audit, and scoring commands.

As a release check, the archived components were reassembled with these entry points: all 17,100 task identities and their groups, instances, families, views, and physical parameters matched the paper metadata, and the reconstructed development set matched all 70 archived groups. One original history per family was also regenerated and passed an ARX prediction check. These checks cover inventory reconstruction and a small CPU path, not a new seven-model reproduction of every result.

Paper experiment artifacts. The experimental archive contains the frozen evaluation inputs and physical references, task and group identities, development/test assignments, fitted local models, prediction arrays, and executed control records. Table 31 distinguishes the full prediction evaluation, formal control evaluation, and clean-FOPDT paired analysis. Task identities and model bindings connect these records to the corresponding histories and information views.

Table 31: **Archived materials underlying the paper experiments.** These inventories overlap and should not be added as independent samples. The table describes retained experimental records; public availability is specified below.

Evaluation component	Archived contents	Scope
Prescribed-input prediction	Input histories and plans, physical reference trajectories, task metadata, prediction arrays, and local fitted models.	17,100 conditions and 135,700 queries per model; seven predictors.
Formal control	Task registration, development/test assignments, fitted-model bindings, executed trajectories, saved plan records, and episode metrics.	6,020 tasks per configuration across nine configurations; 5,594 test tasks and 426 development tasks per configuration.
Paired clean-FOPDT analysis	Matched histories, prediction/reference arrays, executed control sequences, task-level scores, and within-task model-pair comparisons.	940 tasks from 376 test groups; 2,994 eligible prediction probes per model.

Current public availability. The source distribution provides the reference workflow, model adapters, controller implementations, research scripts, frozen generation parameters, and inventory/split checks. The manuscript sources provide aggregate statistics, the model-pair reversal

decomposition, and scripts for regenerating the reported figures and tables. Complete numerical inputs, fitted models, and raw prediction/control outputs are retained separately rather than bundled with the code; the documented formal path generates new arrays and binds new fits before execution. Full regeneration and a new seven-model control evaluation have not been verified end to end. In particular, inventory identity does not certify historical model-runtime identity or identical numerical outputs under another environment. Checkpoint identities and historical dependency evidence remain as specified in Appendix C.2.