PREDICTIVE COMPRESSION FOR STATE–ACCESS ALLOCATION

Anonymous authors
Paper under double-blind review

ABSTRACT

How much state should a prediction head keep, and how often should it read a shared representation? State can preserve predictable future answers across a read gap; new task information can require an earlier read. A whole-gap affine analysis separates these bottlenecks, and controlled synthetic interventions exhibit both. We train one multi-step predictor, compress its future task answers into candidate held states, calibrate their executed risks and deploy the cheapest passing state–read configuration. Across 12 new synthetic worlds, compilation gives a 20.29× cold-preparation speedup over training a complete receiver grid and a 12.11× paired geometric speedup over cost-ordered early-stop receiver search, at 1.038 times their selected normalized resource cost. All 432 price queries yield independently test-passing components at the stated calibration size. A sufficiently trained shared grid can select slightly cheaper budgets on these synthetic tasks, while a simple hold baseline is stronger on the tested frozen video features.

1 INTRODUCTION

When a new prediction head is attached to a shared representation, its designer must choose how much local state it keeps and how often it reads the shared features. State can carry predictable task answers between reads; task information that arrives afterward may require another read. The choice is how to meet an error target with both resources accounted for.

We study a periodic head that reads a fixed-width packet every g steps, keeps d FP32 state coordinates, and answers at every step. It can process a packet nonlinearly when read; between reads its whole gap of answers is affine in the held state. The head cannot reuse uncharged past packets or activations. A video location head that fetches a shared frame feature only on scheduled frames is one example of the interface. We evaluate controlled generated packets and a frozen video representation.

Future predictions as state, low-rank forecast compression, and calibrated risk selection have established foundations (Littman et al., 2001; Barratt et al., 2021; Shlezinger et al., 2019; Angelopoulos et al., 2025). Here we reuse one task forecast at a fixed packet interface to choose state capacity and read interval together, compile a deployable head for each candidate, and measure its preparation-time/resource-cost tradeoff.

Figure 1 shows the resource choice in one preselected synthetic world. With little new information, two more state coordinates make a longer read interval feasible; with more new information, even large states cannot support that interval. Proposition 1 separates these effects for the ideal affine class, and controlled interventions test both regimes (Figure 4).

Our method learns one predictor of the task's next 16 answers. It compresses those future answers into candidate held states, evaluates each constructed component on independent calibration episodes, and deploys the lowest-cost passing state/read pair. Its learned forecast is reused across candidate budgets, so deploying a component does not require training a receiver for every pair. Independent receiver fits test whether a new initialization also realizes the chosen budget. Figure 2 separates preparation from execution.

The main measured result is a 20.29× cold-preparation speedup over a full receiver grid on new synthetic worlds; a matched follow-up against cost-ordered early-stop receiver search retains a

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

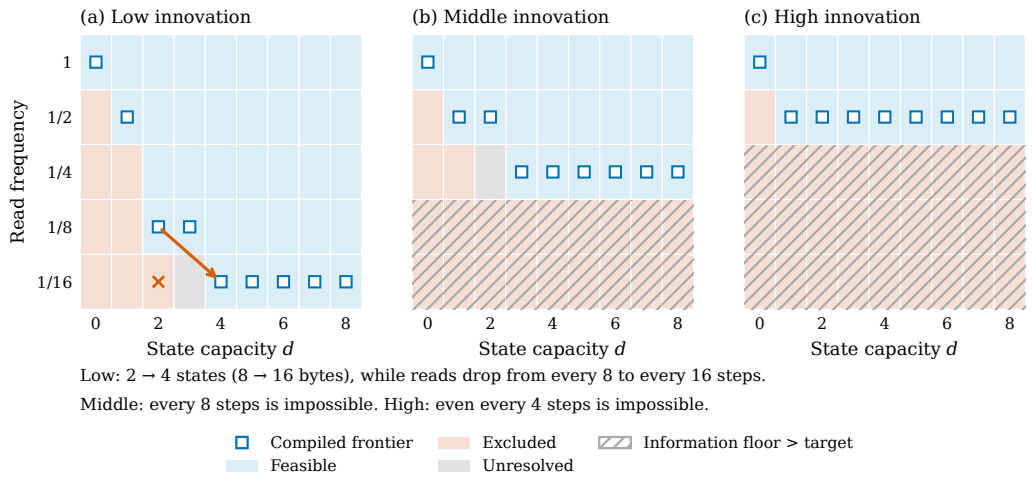


Figure 1: Same task, different new-information rates in a preselected synthetic world. Each cell uses the same 36-byte read packet and target MSE 0.04. Outlined squares mark the compiled predictor's certified frontier. Blue cells have a calibrated feasible compiler; peach cells are excluded for the affine-gap class; gray cells remain unresolved. Hatching marks a source information floor above the target. At low innovation, $(2, 8) \to (4, 16)$ trades eight bytes of state for half as many reads; higher innovation rules out long read gaps regardless of state. This is a feasibility exchange; it lowers J_λ only for $\lambda > 4$. Appendix F gives the bounds and numerical qualification.

$12.11\times$ paired geometric speedup at the same 1.038 selected-resource-cost ratio. Future-answer compression beats specified current-feature compressors; a sufficiently trained shared grid can choose slightly cheaper budgets. On tested frozen video features, sample-and-hold is more accurate at no greater cost. These results connect the state/read phenomenon to a deployable method that reuses one predictor.

2 A COMPONENT WITH TWO RESOURCE CHOICES

Let $(M_t, Y_t)_{t=0}^{T-1}$ be an exogenous packet/target process ($T = 32$ here). A component reads the fixed-width packet only at $t = 0, g, 2g, \ldots$, but predicts $\widehat{Y}_t$ at every step. Past packets, writer activations and outputs cannot be retained outside charged state. For a generic head h, its task risk is

$$R(h; d, g) = \mathbb{E}\left[T^{-1} \sum_{t=0}^{T-1} (Y_t - \widehat{Y}_t)^2\right]. \tag{1}$$

State capacity d counts persistent FP32 coordinates separately from parameters and read-time scratch space. The synthetic packet has nine FP32 values (36 bytes), giving the normalized selection cost

$$J_\lambda(d, g) = d/8 + \lambda/g, \qquad d \in \{0, \ldots, 8\}, \quad g \in \{1, 2, 4, 8, 16\}. \tag{2}$$

Here λ prices packet reads relative to held coordinates. This logical proxy does not measure network transport, encoder work, energy, or latency. The separate video study has a wider packet and its own within-study cost normalization.

Output classes. At a read, an arbitrary nonlinear writer maps the packet to at most d held coordinates $f(m)$. For unseen offsets $j = 1, \ldots, g-1$, the phase output is $b_j + B_j f(m)$; thus the complete gap output lies in one at most d-dimensional affine subspace. We call this $\mathcal{F}_{\text{aff}}$. The read-step head may use m directly. Clipping the writer or stored state can be absorbed into f. Some comparators instead clip the *final* output, $\text{clip}(b + Bf(m), -2, 2)$, giving $\mathcal{F}_{\text{clip-aff}}$. It obeys the same read schedule, but its outputs need not lie in one affine subspace. Spectral-tail claims below concern $\mathcal{F}_{\text{aff}}$; executed-risk upper bounds and lawful-information floors can cover either class.

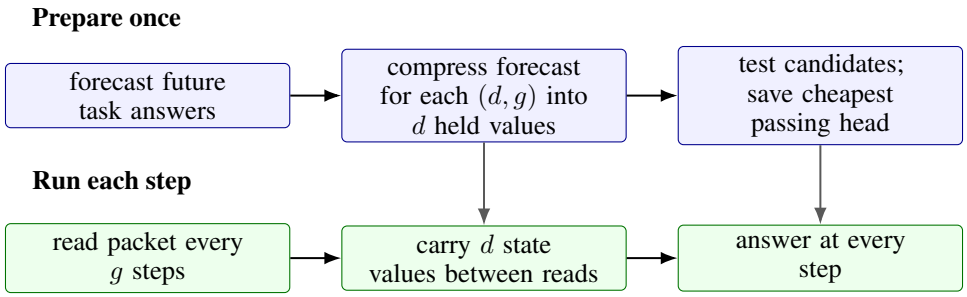


Figure 2: One future-answer forecast is compressed for each candidate (d, g) into the d values held by its runtime head. Calibrated tests select the cheapest passing head; it reads every g steps and answers at every step.

Unread information. Let $\mathcal{I}_g$ contain the scoring time and all packets lawfully read through that time. With $\mu_g^{\text{hist}} = \mathbb{E}[Y \mid \mathcal{I}_g]$ for a uniformly sampled scoring time, conditional-expectation orthogonality gives

$$R(h) = \mathbb{E}(Y - \mu_g^{\text{hist}})^2 + \mathbb{E}(\mu_g^{\text{hist}} - \widehat{Y})^2 \geq \mathbb{E}(Y - \mu_g^{\text{hist}})^2. \tag{3}$$

This floor applies to every legal component under the same exogenous source. A floor conditional on the latest packet alone applies to the full lawful history only when that packet is sufficient; the controlled Markov source has this property at the ideal level.

Saved predictable content. For one read period define

$$Z_g = g^{-1/2}(Y_{t+1}, \ldots, Y_{t+g-1})^\top, \quad v_g(m) = \mathbb{E}[Z_g \mid M_t = m], \quad K_g = \text{Cov}(v_g(M_t)), \tag{4}$$

and $\nu_g^{\text{gap}} = \mathbb{E}\|Z_g - v_g(M_t)\|^2$.

Proposition 1 (Ideal affine gap boundary)**.** *For squared loss and any nonlinear writer whose complete gap output lies in an affine subspace of dimension at most d, the gap risk is at least*

$$\nu_g^{\text{gap}} + \sum_{i>d} \lambda_i(K_g), \tag{5}$$

where eigenvalues are descending. Equality holds in the ideal class when the conditional-mean projection, its mean offset and all phase readouts are implementable within the stated budgets.

The first term is information absent at the read; the tail is predictable variation omitted by finite state. Each extra ideal coordinate saves the next eigenvalue at the same g; changing g changes both terms. The proof is conditional-expectation orthogonality and optimal affine rank-d approximation (Appendix A). The proposition concerns gap output; learned read-step error must still be counted in complete risk. The implementation projects an uncentered second moment with an origin-based code, while the proposition uses centered covariance and permits an offset. Uniform initial phases make the true future-mean offset zero in the controlled sources; finite-sample and prediction errors are handled in the separate class certificate of Appendix B.

3 From a future forecast to a resource choice

The ideal spectrum explains which future variation would have to be saved. It does not tell us the risk of a finite trained network. Our allocator therefore builds an actual component for each candidate resource pair and calibrates *that component's* complete task loss.

Learn and compress in the controlled study. For each task, a neural forecaster N learns 16 scalar heads from the current nine-number packet: the current target and the following 15 targets. In the controlled sources, F0 includes the current target in that packet, whereas F1 requires it to be predicted. The writer is a $9 \to 32$ ReLU $\to 16$ network with a bounded output. The future heads are projected to a rank $r = \min(d, g - 1)$ subspace and written as FP32 state. A phase head reads

that state at each unseen offset. The separately fitted A receiver clamps its state, but the frozen N projection compiler does not clamp the projected coordinates or final gap outputs; its calibrated risk includes their executed values. The sampled projection in this synthetic implementation uses an uncentered second moment of the learned future forecast. This is a construction choice, not an assertion that its eigenvalues equal the ideal covariance spectrum in Proposition 1. An ordinary linear forecast is also evaluated as a prediction baseline.

Calibrate the constructed risks. After predictor and projection selection, n independent 32-step episodes per task are reserved for certificate calibration. The original allocation study uses $n = 524{,}288$; the complete preparation comparison uses $n = 65{,}536$. For each physical compiled cell, define $\widehat{R}_N(d, g)$ from its episode losses and a simultaneous one-sided upper bound $U_N(d, g) \equiv U(d, g)$. Bounded episode losses and an empirical Bernstein inequality give, for a frozen cell, a radius of the form

$$U(d,g) = \widehat{R}_N(d,g) + \sqrt{\frac{2\widehat{V}_{d,g}\log(2/\delta)}{n}} + \frac{7M_{d,g}\log(2/\delta)}{3(n-1)}, \tag{6}$$

where $\widehat{V}_{d,g}$ is the unbiased episode-loss variance and $M_{d,g}$ is a valid range length fixed before calibration. The synthetic sources have $|Y| < 2$; unclipped paths use a conservative 2.1 envelope. For a positive-state compiler, let $\Pi = UU^\top$ be its frozen future-head projection and put $h_j = 2\|\Pi_{j,:}\|_1 + 10^{-4}$; for a trained receiver with state in $[-1, 1]^d$, put $h_j = \|B_{j,:}\|_1 + 10^{-4}$. The executed, unclipped gap outputs obey $|\widehat{Y}_{t+j}| \le h_j$. The code uses

$$M_{d,g} = g^{-1}\left(C_0 + \sum_{j=1}^{g-1}(2.1 + h_j)^2\right) + 10^{-12}, \quad C_0 = \begin{cases} 0 & \text{F0,} \\ (4.1001)^2 & \text{F1.} \end{cases} \tag{7}$$

The zero-state endpoints use the corresponding conservative read-step and zero-output range; their exact rule is given in Appendix C. Projection matrices and receiver weights are frozen on independent lanes before these bounds and calibration losses are computed. The implementation also checks numerical qualifications and reserves a fixed family of at most $500{,}000$ one-sided events per phase with $\delta = 10^{-7}$, yielding a familywise budget of at most 0.05 for each phase. Predictor selection, calibration, receiver validation and final test use separate episode lanes. An upper bound on a *specific compiled construction* is a feasible-cell certificate when $U(d, g) \le q$; it is not a lower bound for every possible network at that budget.

Class exclusions are separate from selection. The source information floor applies to every legal component when the current packet suffices for the lawful read history. For the zero-mean synthetic sources, a qualified spectral-tail certificate can also exclude an entire whole-gap affine cell. Neither is inferred from one failed network fit. The allocator below selects using constructed-risk upper bounds; Appendix B gives the class-level lower rule, its quantifiers and numerical qualifications.

Choose; then check independent training. At target $q = 0.04$ and a prespecified λ, the allocator selects

$$(\widehat{d}, \widehat{g}) \in \operatorname*{argmin}_{(d,g):U(d,g)\le q} J_\lambda(d, g), \tag{8}$$

breaking ties by smaller d then smaller g. All 432 task/price choices can be reconstructed from frozen predictions and this rule, before independent receiver training.

The compiled N component is the directly deployable output. Separately training an affine receiver A from independent initializations tests whether another fit realizes the same budget. It uses the same packet, writer width, FP32 state and output interface, with a different objective; both seeds must pass on held-out test episodes at a selected pair.

The original allocation audit uses two independent A seeds per cell; the later cold-path study trains one seed per cell in each of three task repeats. Each grid selects its cheapest passing cell under its stated validation rule. Development-fixed single-axis policies still adapt the other resource by task. The cold-path study times direct compilation, shared supervision, and full-grid training. Complete recipes and split roles appear in Appendix C.

4 EXPERIMENTS: PREPARATION, MECHANISM, AND LIMITS

Controlled tasks. Two source families map circular latent modes through the same invertible eight-scalar representation inside a nine-scalar packet. F0 supplies the current target at a read and adds fresh observation noise; F1 instead adds latent-state innovations, places nuisance noise in the ninth coordinate, and requires a learned current estimate. Each of 24 independent base worlds has one, two, or four active modes and three paired innovation levels. An active mode spans at most two future-mean directions, so the original construction has predictive rank at most eight, the largest state budget. Worlds, rather than paired tasks, training seeds, or price queries, are the uncertainty units. All synthetic risk tests use independent 32-step episodes. Full source and training specifications are in Appendices C and E.

4.1 A DEPLOYABLE BUDGET WITH LESS PREPARATION

On 12 new synthetic worlds, each of five original paths prepares both $\lambda \in \{1, 4\}$ choices per task. Three repeats per paired task give 216 complete runs and 432 price queries. Single-thread cold timing includes data, fitting, projection where used, calibration, selection and saving; the independent research test is excluded. We denote direct future-answer compilation by F. With 65,536 calibration episodes, direct compilation takes 1.470 seconds per task versus 29.827 for the independently trained full receiver grid: a prespecified 20.29× paired geometric preparation speedup (95% 12-world bootstrap interval $[20.17, 20.40]$). Both paths deliver 432/432 independently test-passing choices; F's selected resource cost is 1.038 times the full-grid cost (world interval $[1.014, 1.068]$). The long-trained shared grid chooses slightly cheaper budgets ($J = 0.904$ versus F's 0.945), while taking 26.040 seconds per task. Figure 3 shows the measured preparation–resource comparison.

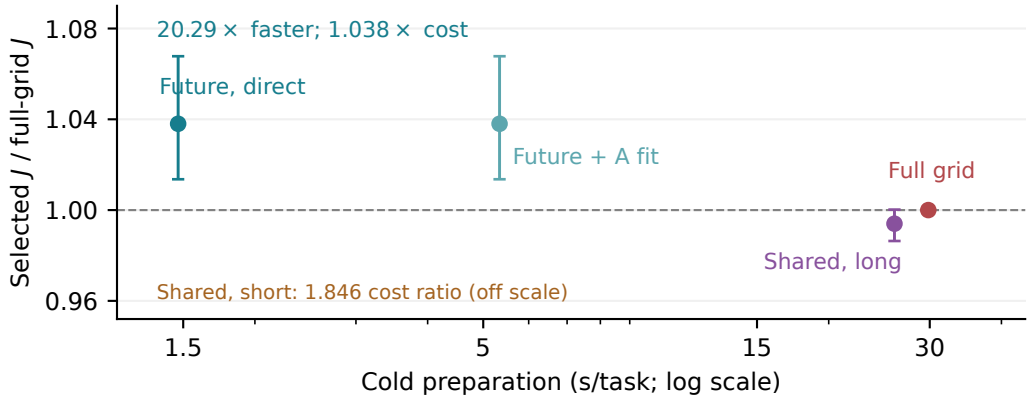


Figure 3: Complete cold preparation versus selected normalized state/read cost for five preparation paths at 65,536 calibration episodes. Costs are paired world means relative to the full receiver grid; bars are 95% world-bootstrap intervals. The short-trained shared grid's cost ratio (1.846) lies above the zoomed axis. All five paths deliver and test-pass every price query.

A cost-ordered receiver search trains only until the first passing cell for each requested price, reusing visited cells. On the same 12 previously exposed worlds, a same-session paired run takes 20.009 seconds/task for this early-stop search versus 1.443 for F: a 12.11× geometric speedup (95% 12-world interval $[10.12, 14.32]$). Search trains 12.98 of 19 positive-state cells on average, yet reproduces all 216 full-grid task choices. Its selected receiver checkpoints and F's predictor and projection arrays match those from the original full-grid run, allowing the same independent test readout: both paths pass all 432 price queries, and F's selected-cost ratio remains 1.038. The timing ratio compares only the paired runs within one session on these exposed worlds. Search order and checkpoint checks are in Appendix E.

Calibration size affects delivery: with 16,384 episodes F finds no certified cell in 216/432 queries, whereas at 65,536 all paths deliver. The 20.29× figure therefore describes this operating point and

the full-grid comparison. Absolute seconds are scoped to their CPU sessions; Appendix E gives the complete path table and calibration sweep.

4.2 When state helps and when a read is needed

In 24 new worlds, increasing state from $d = 1$ to 7 at fixed $g = 8$ reduces mean MSE by 0.1672 in F0 and 0.1666 in F1 under low innovation. At high innovation, shortening the gap from $g = 8$ to 2 at fixed $d = 1$ reduces MSE by 0.1595 and 0.1812, respectively. All four adjusted world-bootstrap lower endpoints exceed the prespecified 0.01 practical scale. At high innovation the source information floor at $g = 8$ exceeds $q = 0.04$ in all 24 worlds per family, while both $(1, 2)$ receiver seeds pass; at low innovation both $(7, 8)$ seeds pass where the ideal affine $(1, 8)$ reference exceeds q. Figure 4 shows the interventions. The source changes innovation location and read-step target access together, so the two families are complementary stresses rather than a one-variable ablation.

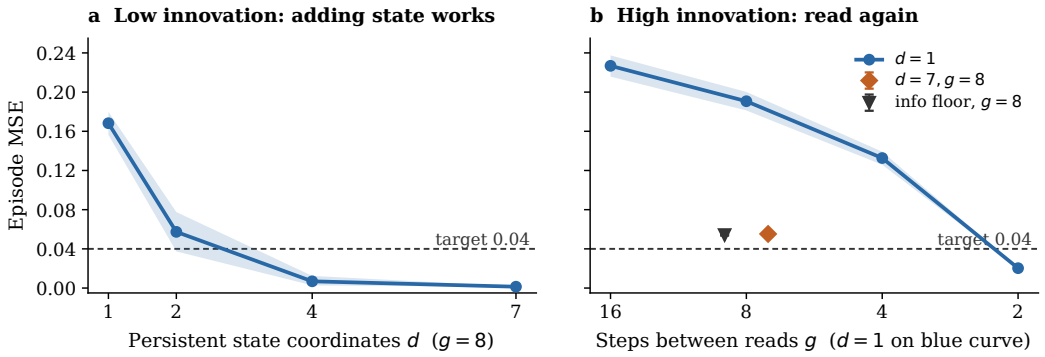


Figure 4: Two bottlenecks in 24 independent worlds. More state at fixed $g = 8$ repairs low-innovation error (left); under high innovation, an earlier read succeeds where more state at $g = 8$ cannot (right). Points average F0/F1 and two seeds within world; uncertainty is pointwise 95% world bootstrap. The source floor is a numerical reference. Both displaced right-panel marks have exactly $g = 8$.

In the original allocation study, all 432 compiled choices pass final risk tests. For the 288 noncheap price queries, 285 selected pairs match an independently trained full receiver grid, and both new receiver seeds pass at every selected pair. The three mismatches reach a worst single-query cost ratio of 1.5. At $\lambda = 4$, choosing both axes saves 13.9% in F0 and 13.7% in F1 against a policy that fixes capacity but still adapts read frequency by task. The cheap-read $\lambda = 0.25$ queries all select direct current access, so joint adaptation has no gain there. These comparisons establish budget quality within the finite grid; the detailed price table, three disagreements, and a concrete capacity-for-frequency exchange are in Appendix F.

4.3 Why compress future task answers?

On 24 new worlds from the original source formulas, three compilers share one trained predictor, state counts, read rules, calibration and test lanes. They compress either future task predictions (F), the current packet by packet PCA (P), or current hidden features by feature PCA (H). F selects a lower-cost passing cell than P and H in all 288 task/price comparisons; its world-mean cost saving over H ranges from 32.5% to 40.5%. The three executed comparators use the same final output clip and belong to $\mathcal{F}_{\text{clip-aff}}$.

A stronger control shares a supervised writer over all 19 physical cells. With the same total 2,000 updates as the future predictor, its chosen budgets are costlier; with 38,000 updates (roughly 2,000 per cell), it ties F in 70–71 of 72 tasks per source/price row and wins the remaining one or two. Thus the observed advantage is reuse of predictive training across configurations, within the measured preparation tradeoff, rather than an exclusive representation capability. A linear future predictor

also loses allocation quality at high read price. Full comparator tables, training exposures, and risk checks are in Appendix F.

4.4 A HARDER SPECTRUM AND RECEIVER REALIZATION

An eight-harmonic target removes the original rank-eight cutoff: at $g = 16$ its numerical future-mean rank is 11–15 across the two sources. In 24 new worlds per family, every compiled selected component passes its held-out risk bound, and 47/72 F0 and 48/72 F1 choices per price retain $g > 1$. Independently initialized receivers fail the two-seed check in up to 6/72 selected tasks per source/price row. On 24 further high-harmonic worlds, extending receiver training from 2,000 to 20,000 updates improves MSE by 0.00771 (adjusted world interval $[0.00694, 0.00857]$) and raises both-seed selected-cell passes from 64/69 to 69/69. A direct zero-gradient mapping of the origin-based compiler is legal in all 117 checked cells and reproduces outputs within 7.2×10^{-7} in FP32. These observations separate an available bounded representation from whether a random fit finds it; the detailed fixed-cell failures remain in Appendix E.

4.5 AN ADVERSE FROZEN-VIDEO TEST

On DAVIS 2017, a frozen ImageNet ResNet-18 supplies a 4,096-coordinate packet per frame; the target is foreground centroid from withheld masks. Forty videos train models, ten choose checkpoints, ten select budgets, and 30 official validation videos are held out. The video cost is $J_\lambda = d/32 + \lambda/g$. Budget selection uses video-equal empirical MSE on its ten videos, rather than the synthetic episode certificate. No evaluated F or Direct-A configuration meets the prespecified $q = 0.01$ or 0.02 target across three seeds. A $q = 0.04$ successor was set after inspecting selection videos and before opening the 30 held-out videos.

Table 1: Held-out video performance at adapted $q = 0.04$, $\lambda = 4$. Costs list three seeds; MSE equally weights 30 videos, then seeds.

Method	Selected J across seeds	Video MSE
Future compression F	.5625, .5625, .5625	.0360
Sample-and-hold	.5625, .5625, .3125	.0308
Direct-A grid	.5625, .625, .625	.0338
Long shared grid	1.0625, 1.0625, .625	.0341

Sample-and-hold stores the current task estimate and is more accurate than F by 0.00519 MSE over paired videos (95% video-bootstrap interval $[0.00129, 0.00941]$), at no greater cost in any seed. F's held-out mean is below 0.04, but its sequence-bootstrap upper endpoint is 0.051–0.052; this is empirical success, not population-risk certification. A same-$g = 8$ post-hoc diagnosis separates the executed models: holding a separately trained current head gives 0.02918 MSE; holding N's current head, 0.03315; N's uncompressed future predictions, 0.03812; and compressed F, 0.03598. The executed MSE difference $R(F) - R(E_{\text{hold}}) = 0.00680$ telescopes as $0.00397 + 0.00497 - 0.00214$ across these four outputs; this is a descriptive difference, not a causal error decomposition. Compression improves N's whole-window forecast here, while its current estimate and future adjustments lag the simple hold. Figure 5 shows the offset pattern; the comparison does not estimate the Bayes future mean. At equal g, the unread-innovation term cancels from the risk difference between lawful outputs, so unpredictability alone does not explain the advantage of hold. Full controls and the selection rule are in Appendix E.

4.6 SCOPE OF THE AFFINE BOUNDARY

A correctly output-clipped nonlinear receiver crosses the ideal affine minimum by the prespecified 0.005 margin in seven task/cells on five new F1 worlds, but is worse than an output-clipped affine control on average by 0.00993 MSE. This is a class-boundary result, not an average allocation advantage. A fixed-weight increase in read-time iterations from one to eight produces no 0.005-scale improvement in its tested writer. Appendix F reports the risk brackets, full controls, numerical gate slack, and execution history.

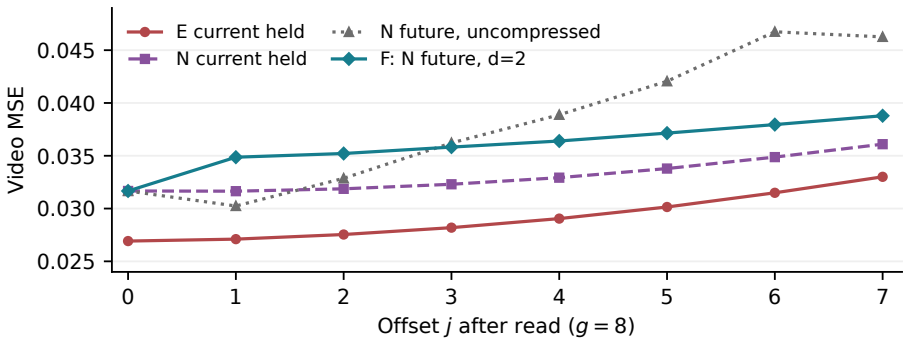


Figure 5: Post-hoc diagnosis at fixed $g = 8$ on 30 already exposed held-out videos and three frozen seeds. Points average within video, then across videos and seeds. E is the separate current-task head; N is the multi-step predictor. F uses $d = 2$; hold controls store their current estimate. The curves are descriptive and do not estimate Bayes-predictable change.

5 Related work and interpretation

Multirate modules and intermittent features. Clockwork RNNs update hidden modules at different periods (Koutník et al., 2014); Deep Feature Flow uses sparse key frames and propagated features (Zhu et al., 2017). These motivate refresh scheduling; here we price reads of a fixed packet and measure preparation of the selected component. Intermittent-observation Kalman filtering studies missed inputs in linear-Gaussian estimation (Sinopoli et al., 2004); we select periodic, task-specific state/read budgets.

Predictive state and compression. Predictive state representations use future tests (Littman et al., 2001). Principal-component reconstruction (Baldi & Hornik, 1989) and Low Rank Forecasting's compact future decoder (Barratt et al., 2021) supply the compression tools. We compile one task forecast into directly deployable components for state/read pairs. Task-based quantization separates estimation and compression errors while targeting task information (Shlezinger et al., 2019).

State Predictive Information Bottleneck learns a compact, time-lagged predictive coordinate for molecular dynamics (Wang & Tiwary, 2021). Bottlenecked Transformers periodically consolidate and rewrite a language model's KV cache to retain predictive information for reasoning (Oomerjee et al., 2026). Our fixed-packet setting makes the jointly selected FP32 state count and periodic read interval explicit. Joint observation and compression constraints also have a sampling-theory precedent: the sampled-Wiener distortion-rate function trades uniform sampling rate and coding bitrate against squared reconstruction error (Kipnis et al., 2019). We use conditional expectation and spectral truncation to analyze fixed-precision persistent coordinates, periodic reads of a fixed-width packet, and loss on a task's future answers under a whole-gap affine output restriction. The empirical question is whether this construction selects a passing, directly deployable (d, g) with less preparation than a receiver grid, and how reliably independent fitting realizes that budget.

Calibrated risk selection. Empirical Bernstein bounds provide data-dependent risk intervals for frozen bounded-loss functions (Maurer & Pontil, 2009). Learn then Test gives a more general way to choose from a finite parameter family while controlling risk without knowing Bayes risk (Angelopoulos et al., 2025). Our upper-bound feasibility step inherits that logic. The additional, experimentally tested question is whether the task-forecast construction gives useful directly deployable (d, g) choices in this fixed family. Separately trained receivers provide an empirical realization check, not the deployable certificate.

6 Discussion

The observed state/read tradeoff has two causes: state preserves predictable future task content, while a new read supplies information that arrived after the last one. The spectral argument expresses

this distinction for whole-gap affine outputs under its stated offset and implementation conditions, and the controlled interventions exhibit both remedies. Our deployable result uses a future-answer predictor to choose the budget: on the controlled sources, it avoids most full-grid preparation while selecting passing components with a small normalized resource premium.

The comparisons delimit method advantage. A long-trained shared grid can match or slightly improve budget cost. High-harmonic tasks expose receiver-fitting failures even when a bounded representation is available. On held-out DAVIS videos, a simple task-matched hold is more accurate at no greater selected cost; the fixed-gap diagnostic locates part of the gap in the learned current and future heads, without identifying the Bayes future mean. The video's adapted $q = 0.04$ selection is empirical and its 30-video interval does not certify population risk below that target. Final-output-clipped and nonlinear gap controls lie outside the affine spectral-tail claim, while retaining the lawful-information limit and executed-risk checks.

The paired time ratios describe preparation within their respective CPU sessions. On the exposed synthetic worlds, cost-ordered receiver search has a 12.11-fold paired geometric time ratio to compilation; this does not establish the best possible receiver-search time. The cost J_λ prices held coordinates and packet reads; encoder work, physical communication, energy, and adaptive or action-dependent reading remain separate questions.

Reproducibility and materials

The anonymous artifact accompanying this submission candidate provides stored-result replay for the synthetic preparation and calibration comparisons and the adverse video comparison, selected synthetic checkpoints, compiler source, and the first result figure's source and data. Its README gives commands and scope; the appendix gives proofs and experiment specifications. These checks replay retained results and include a finite compiler smoke test; they do not rerun training. DAVIS images, extracted features, and third-party encoder weights are excluded, with official setup sources listed in the artifact.

AI Use Statement

OpenAI Codex assisted with problem and hypothesis refinement, literature screening, mathematical derivations, experiment design, implementation, analysis and interpretation, figure preparation, and manuscript drafting and editing, and supplementary artifact preparation. Reported numerical results are tied to preserved code and run artifacts. The authors take responsibility for the final methods, results, claims, and text.

References

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662, 2025. doi: 10.1214/24-AOAS1998. URL https://arxiv.org/abs/2110.01052.

Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989. doi: 10.1016/0893-6080(89)90014-2. URL https://www.sciencedirect.com/science/article/pii/0893608089900142.

Shane Barratt, Yining Dong, and Stephen Boyd. Low rank forecasting, 2021. URL https://arxiv.org/abs/2101.12414.

Alon Kipnis, Andrea J. Goldsmith, and Yonina C. Eldar. The distortion-rate function of sampled Wiener processes. *IEEE Transactions on Information Theory*, 65(1), 2019. doi: 10.1109/TIT.2018.2878446. URL https://arxiv.org/abs/1608.04679.

Jan Koutník, Klaus Greff, Faustino Gomez, and Jürgen Schmidhuber. A clockwork RNN. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings*

of Machine Learning Research, pp. 1863–1871, 2014. URL https://proceedings.mlr.press/v32/koutnik14.html.

Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems*, volume 14, 2001. URL https://proceedings.neurips.cc/paper/2001/hash/1e4d36177d71bbb3558e43af9577d70e-Abstract.html.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein bounds and sample variance penalization. In *Proceedings of the 22nd Annual Conference on Learning Theory*, 2009. URL https://www.cs.mcgill.ca/~colt2009/papers/012.pdf.

Adnan Oomerjee, Zafeirios Fountas, Haitham Bou-Ammar, and Jun Wang. Bottlenecked transformers: Periodic KV cache consolidation for generalised reasoning, 2026. URL https://arxiv.org/abs/2505.16950.

Nir Shlezinger, Yonina C. Eldar, and Miguel R. D. Rodrigues. Hardware-limited task-based quantization. *IEEE Transactions on Signal Processing*, 67(20):5223–5238, 2019. doi: 10.1109/TSP.2019.2935864. URL https://arxiv.org/abs/1807.08305.

Bruno Sinopoli, Luca Schenato, Massimo Franceschetti, Kameshwar Poolla, Michael I. Jordan, and Shankar S. Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004. doi: 10.1109/TAC.2004.834121. URL https://web.eng.ucsd.edu/~massimo/Papers_files/Kalman.pdf.

Dedi Wang and Pratyush Tiwary. State predictive information bottleneck. *The Journal of Chemical Physics*, 154(13):134111, 2021. doi: 10.1063/5.0038198. URL https://arxiv.org/abs/2011.10127.

Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2349–2358, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Zhu_Deep_Feature_Flow_CVPR_2017_paper.html.

A Proof and qualifications for the ideal boundary

Here we separate the information statement, which applies to any legal component under an exogenous source, from the affine-subspace statement, which depends on the complete gap decoder.

Lawful information. Select a scored time uniformly and write Y for its target and $\mathcal{I}_g$ for all information legally available by that time under the chosen read schedule, including the time itself. Every legal output H_F is measurable with respect to $\mathcal{I}_g$ (and any independent random seed). With $\mu = \mathbb{E}[Y \mid \mathcal{I}_g]$, orthogonality gives

$$\mathbb{E}(Y - H_F)^2 = \mathbb{E}(Y - \mu)^2 + \mathbb{E}(\mu - H_F)^2. \tag{9}$$

Thus $\mathbb{E}(Y - \mu)^2$ is an information lower bound independent of d. It does not say that all errors above it are due to state dimension: architecture restrictions, parameter limits and optimization may also contribute. If the component affects future packets or targets, the compared source distribution changes and this common-distribution decomposition cannot by itself justify a resource intervention.

Affine read-gap output. Fix g and let Z_g be equation (4). For any writer $f : M_t \mapsto \mathbb{R}^d$ and phase readouts, stack the future outputs into $H = b + Bf(M_t)$. Its image lies in one affine subspace of dimension at most d. Conditioning on M_t yields

$$\mathbb{E}\|Z_g - H\|^2 = \underbrace{\mathbb{E}\|Z_g - v_g(M_t)\|^2}_{\nu_g^{\mathrm{gap}}} + \mathbb{E}\|v_g(M_t) - H\|^2. \tag{10}$$

For a fixed affine subspace, the last term is minimized by projecting $v_g(M_t)$ onto it. Optimizing the offset sets it to the mean of v_g; optimizing the d directions takes the leading eigenvectors of K_g.

The remaining squared distance is $\sum_{i>d} \lambda_i(K_g)$, proving the lower bound in equation (5). In the unconstrained real-valued class, an achieving code is

$$f(m) = U_d^\top (v_g(m) - \mathbb{E}v_g), \qquad H = \mathbb{E}v_g + U_d f(m), \tag{11}$$

where columns of U_d are leading eigenvectors. Equality for a specified finite neural family additionally requires implementing the conditional mean, projection, offset and phase readouts within its budget. Approximate or finite-precision implementation gives a legal upper construction only after its errors are controlled or measured. If $g = 1$, the future-gap vector is empty; the full risk is the read-step risk and need not vanish (in F1 it does not).

Where finite implementation error enters. For a fixed orthogonal projector Π and any complete gap output $H(M_t)$ in its image, two orthogonal decompositions give the exact identity

$$R_{\mathrm{gap}}(H) = \nu_g^{\mathrm{gap}} + \mathbb{E}\|(I - \Pi)v_g\|^2 + \mathbb{E}\|\Pi v_g - H\|^2. \tag{12}$$

For the idealized executed N compiler, $H = \Pi\widetilde{V}$ with $\widetilde{V}$ its forecast of v_g, so the final term is $\mathbb{E}\|\Pi(v_g - \widetilde{V})\|^2$. The middle term is the true predictive content lost outside the chosen subspace; its excess over the best rank-d spectral tail measures subspace selection error. The final term charges forecaster error within the retained subspace. This is an accounting identity, not a new optimality theorem. It assumes an exact orthogonal projection and a gap output in its image; FP32 rounding, state clamps and any final output clip must be included as implementation residuals or measured directly. The P26 N compiler uses an uncentered, origin-based FP32 projection and does not apply the trained A receiver's state clamp. Its full executed risk, including the F1 read-step prediction, is what the upper bound calibrates.

The identity also makes a sufficient selection-margin condition explicit. Let R_c^* be the ideal affine risk at grid cell c, $e_c = R_c^{\mathrm{comp}} - R_c^* \geq 0$ its actual construction excess, and $b_c = U_c - R_c^{\mathrm{comp}} \geq 0$ on the simultaneous calibration event. If the cheapest ideally feasible cell c^* obeys $R_{c^*}^* + e_{c^*} + b_{c^*} \leq q$ and every cell ordered before c^* by cost and the stated tie rule has $R_c^* > q$, the upper-bound selector returns c^*. A small forecast error alone does not ensure this: margins to q, projection error and calibration width all matter. We do not claim these sufficient margins hold in every P26 task.

Sufficiency and the read step. The latest-packet $v_g(m)$ concerns components that use the current packet as their read-time writer input. If a broader component can preserve earlier information in charged state, a packet-based ν_g is a lower bound for that broader class only when the current packet is conditionally sufficient for the relevant future mean relative to the full lawful history. The invertible current-state packet and Markov source in our synthetic idealization meet this condition. The risk of a complete g-step period adds the read-step contribution to the scaled gap contribution. In particular, F1's current head has nonzero learned error and is included in its empirical full-period compiler risk $U(d, g)$. The exact ideal-source F1 evaluator separately computes the conditional risk at the read step; it does not set that trained-head error to zero.

B What the frozen lower certificates exclude

The ideal spectral formula and the finite-sample exclusion rule use different objects. For one read gap, let $V = v_g(M_t)$ be the true conditional future-mean vector and $\widetilde{V}$ the frozen forecaster's future-output vector, both scaled by $g^{-1/2}$. In our F0/F1 sources, uniform initial phases and symmetric independent innovations give $\mathbb{E}V = 0$. Therefore the best affine rank-d approximation error of V equals its best *origin-linear* rank-d approximation error. If $s_d(X)$ denotes the latter error's L^2 norm, the distance to the set of rank-d linear reconstructions is 1-Lipschitz:

$$s_d(V) \geq [s_d(\widetilde{V}) - \|V - \widetilde{V}\|_{L^2}]_+. \tag{13}$$

The equality of affine and linear optima here concerns the *true* mean V; the learned $\widetilde{V}$ need not be exactly centered. Without the zero-mean source condition, an uncentered moment tail alone would not certify the full affine class.

The P25 frozen calibration stores a simultaneous upper bound Q_g^{up} on the uncompressed forecaster's per-period future loss. Since the supplied source floor ν_g is its conditional-variance term,

conditional-expectation orthogonality gives $\|V - \widetilde{V}\|_{L^2}^2 \le (Q_g^{\mathrm{up}} - \nu_g)_+$. For P26, which withholds ν_g, the conservative substitute is $\|V - \widetilde{V}\|_{L^2}^2 \le Q_g^{\mathrm{up}}$; its F1 Q_g^{up} also charges the read-step prediction error. The empirical uncentered second-moment matrix of $\widetilde{V}$ has ordered eigenvalues $\widehat{\lambda}_i$. With $p = g - 1$ and a simultaneous entrywise deviation radius b_g, the frozen code uses

$$\underline{T}_d(g) = \left[\sum_{i>d} \widehat{\lambda}_i - (p-d)_+ p b_g \right]_+ \tag{14}$$

as a conservative lower bound on $s_d(\widetilde{V})^2$. It also subtracts $\zeta_{d,g} = 32\gamma_r\sqrt{p/g}$, where $r = \min(d, p, 8)$, $\gamma_r = (2r+2)2^{-24}/(1-(2r+2)2^{-24})$, and $\zeta_{0,g} = 0$. Combining these terms with equation (13) yields the P25 rule

$$L_{\mathrm{aff}}^{\mathrm{known}}(d, g) = \nu_g + \left[\sqrt{\underline{T}_d(g)} - \sqrt{(Q_g^{\mathrm{up}} - \nu_g)_+} - \zeta_{d,g} \right]_+^2, \tag{15}$$

and P26's stored blind rule, which withholds ν_g and uses $[\sqrt{\underline{T}_d(g)} - \sqrt{Q_g^{\mathrm{up}}} - \zeta_{d,g}]_+^2$. If the qualified P25 bound exceeds q, the whole affine cell is excluded. Because read-step squared error is nonnegative, a lower bound for the gap remains a lower bound for full-period risk; F0's read step is supplied exactly, whereas F1's is learned.

Figure 1 colors a cell blue if the frozen compiled construction has $U_N \le q$, peach if its P25 class certificate has $L_{\mathrm{aff}}^{\mathrm{known}} > q$, and gray otherwise. Hatching additionally marks $\nu_g > q$, which excludes every state size for the fixed read schedule under this source's sufficient-current-packet assumption. A lower confidence limit for one trained A receiver is *not* substituted for $L_{\mathrm{aff}}^{\mathrm{known}}$. The analytic source floor, spectral quantities and FP32 error allowance were checked against frozen numerical runs, but the FP64 trigonometric source lacks a uniform interval rounding proof. Accordingly, "excluded" here is a source- and affine-class-qualified certificate, not a universal impossibility statement for arbitrary neural gap dynamics.

C Source construction, data separation and statistical unit

The source specifics matter because a newly arriving innovation is precisely what a stored state cannot recover without reading. Each world has four circular modes, $k \in \{1, 2, 4\}$ active; active task coefficients are $\pm 1/\sqrt{k}$. Initial phases are uniform on the circle. A pair of fixed random rotations Q_1, Q_2 surrounds a leaky-ReLU representation map. The rotations and frequencies are shared across the paired source families and three strengths in a base world. The task signal is normalized to variance $1/3$.

In F0, phase evolves deterministically at its world-specific frequency and Y_t mixes the predictable cosine signal with fresh bounded observation noise. The packet includes Y_t and the eight-dimensional representation. In F1,

$$\theta_{t+1,\ell} = \theta_{t,\ell} + \omega_\ell + \sigma u_{t,\ell}, \qquad u_{t,\ell} \sim \mathrm{Unif}[-1, 1], \tag{16}$$

and the packet is the eight-dimensional representation plus one independent nuisance scalar. The target is the normalized active-mode cosine sum. The F0 strength intervals for its observation-noise coefficient are $[0, 0.1]$, $[0.37, 0.38]$, and $[0.42, 0.44]$; F1 uses $\sigma \in [0, 0.10]$, $[0.18, 0.22]$, and $[0.38, 0.42]$. The two families therefore test different combined source/interface contracts; their numerical differences cannot isolate one changed feature as a causal mechanism.

For F1's ideal evaluator, $\rho = \sin(\sigma)/\sigma$ with $\rho = 1$ at $\sigma = 0$. The conditional future mean decays as ρ^j, and the ideal innovation floor for a period is

$$\nu_g^{\mathrm{ideal}} = \frac{1}{3g} \sum_{j=0}^{g-1} (1 - \rho^{2j}). \tag{17}$$

The F1 reference includes the exact conditional current answer at $j = 0$; its learned receiver still incurs current-head error. Since the four active-mode weights have magnitudes $1/\sqrt{k}$, $|Y_t| \le \sqrt{2/3} \sum_{i \in \mathrm{active}} 1/\sqrt{k} = \sqrt{2k/3} \le \sqrt{8/3} < 2$ in F1. For any such target y and any prediction h,

Euclidean projection onto $[-2, 2]$ obeys $(y - \mathrm{clip}(h, -2, 2))^2 \le (y - h)^2$. This holds for every time point and therefore for episode and population MSE. In particular, each valid upper bound on an executed unclipped C checkpoint also upper-bounds its post-clipped version. This is a post-processing risk statement, not a claim that clipped outputs were used during the frozen C training or originally measured. The analytic floor and spectrum were checked numerically with FP64 source operations; the source trigonometry has no uniform interval rounding proof. These ideal values are evaluator aids withheld from the blind allocator. The blind method does not read the latent phase, frequency, number of active modes, innovation strength, rotations, or world seed.

Development bases 10–12 fix the learning recipe. Main bases 300–323 give eight independent worlds per k value. Strength variants share raw random streams within a world, so paired variants are not new independent worlds. Each task has separate episode lanes for forecast fitting, forecast selection, projection, certificate calibration, receiver fitting/selection, configuration validation and final test. Calibration and test each use $524{,}288$ independent 32-step episodes per task lane. Time steps within an episode can depend on one another; the episode is the concentration unit. Cost differences are first computed within each base world across its three strengths, then averaged over 24 worlds. The preserved confidence intervals use 20,000 stratified world-level bootstrap samples by k. Intervals against the stronger fixed-capacity baseline are per-comparison 95%; the separately reported fixed-pair primary comparison uses Bonferroni simultaneous intervals over six family/price conditions.

Executable calibration ranges. The source formula gives $|Y_t| < 2$ for both families: F1 is at most $\sqrt{8/3}$ in magnitude, and F0 is at most $\sqrt{(8/3) + 1} < 2$ by Cauchy–Schwarz over its signal/noise mixture. The original unclipped paths conservatively use a 2.1 target envelope. The bounded forecast and current heads output in $[-2, 2]$. For the original unclipped compiler and affine receiver, equation (7) uses the frozen projection or readout rows from the independent fitting lanes, before calibration examples are opened. For a zero-state endpoint, the code uses $M = (4.1001^2 + (g-1)2.1^2)/g + 10^{-12}$; the exact F0 feedthrough at $g = 1$ instead uses 10^{-12}. These bounds cover the episode-average squared loss because every tested gap divides the 32-step episode. The P27/P28 final-output-clipped controls use $M = 16$ for predictions and targets in $[-2, 2]$; no post-calibration tightening is allowed. The one-sided event probability is 10^{-7}, and at most 500,000 events per phase are reserved. P34 changes only the primary calibration count to 65,536; the loss-range rule and final 524,288 episode test remain the same.

Table 2: Executed original allocation implementation. Counts are per task unless marked otherwise. FP32 models were trained on CPU; empirical moments and the ridge solve use FP64 accumulation. The episode is the risk unit.

Item	Executed specification
Data lanes	N/L: fit 8192 episodes (lane 1), select 1024 (2), projection 8192 (3), calibrate 524288 (4). $A/E/C$: fit 8192 (5), select 1024 (6); configuration validation 524288 (8), final test 524288 (9). Each episode has 32 scored steps.
Forecaster N	$9 \to 32$ ReLU $\to 16$ heads, outputs clipped to $[-2, 2]$; 848 parameters. The 16 heads predict one read step and 15 future steps. F0 fit/selection loss omits the directly supplied current target.
Linear L	Affine $9 \to 16$ ridge forecast (160 coefficients), regularization selected as zero; clipped heads. Same projection, calibration and allocation rule as N.
Receiver A	Same $9 \to 32 \to 16$ writer, followed by a $15 \to d$ linear state map and $(g-1) \times d$ affine-gap phase rows; $848 + d(g + 14)$ parameters. Writer clipped to $[-2, 2]$, state to $[-1, 1]$, each readout row ℓ_1 norm at most 32. The executed final gap output is not clipped.
Scope head C	A plus $d \to 32$ ReLU $\to d$ state transition; $880 + d(g + 79)$ parameters. State is clamped at each gap step. The executed final gap output omitted the required $[-2, 2]$ clip.
Optimization	Adam, learning rate 10^{-3}, 2000 steps; minibatch 128 for N, 64 for $A/E/C$; gradient norm clipped at 1. Choose the lowest selection MSE checkpoint among steps 200,400,. . .,2000. Two independent A seeds per physical cell.

Grid and bound	19 physical positive-state affine cells per task, with padded allocated aliases; $d = 0{:}8$, $g \in \{1, 2, 4, 8, 16\}$. Uncentered future-head projection; N compiler does not clamp its projected state or final gap output. Simultaneous empirical-Bernstein limits use $\delta = 10^{-7}$ per one-sided event.

Follow-up comparison specifications. P27 uses new source bases 500–523, master seed 2026092527, and the same N training and source formulas as P26. Future F projects its 16-head task forecast; packet P projects the current nine-number packet; hidden H projects the 32 ReLU features. P/H use origin-linear lane-3 decoders to the same frozen future heads. All three use the same $[-2, 2]$ final output clip, 524288-episode calibration and independent test lanes, and $\delta = 10^{-7}$ one-sided intervals. P28 uses new bases 600–623 and master seed 2026092528. It reruns F/P/H and trains a 2668-parameter shared writer plus 19 cell heads on 8192 lane-1 episodes, selecting checkpoints by mean grid loss on 1024 lane-2 episodes. Adam uses learning rate 10^{-3}, batch 64, gradient norm clip 1, and a permuted 19-cell cycle. S-total runs 2000 updates; S-per-cell runs 38000; each has ten checkpoint options. All methods calibrate and test on separate 524288-episode lanes. P29 uses new bases 800–823, master seed 2026092529, and the same N/A optimizer, grid, and lanes as P26. Its harmonic target changes the source, not the method recipe; all N choices were frozen before selected-cell A fits. The six-world implementation pilots for each extension are excluded from confirmatory summaries.

D HISTORICAL COST RECONSTRUCTION AND CALIBRATION REPLAY

The P26 logs give only a partial ledger: across 144 tasks, N fits sum to 100 CPU seconds, complete prediction workers to 3808 worker-wall seconds including 524288-episode calibration, and 5472 physical A fits to 8596 CPU seconds. CPU and worker-wall totals are not an end-to-end comparison; that audit trained the full grid. In a retrospective prefix replay, all 288 noncheap choices at 65,536 calibration episodes still select an A cell whose two seeds pass the original test, but only 189/288 choices match the full-calibration pair; at 262,144 episodes, 276/288 match. At 16,384 episodes no F1 cell is certified feasible. These exposed-lane replays are descriptive; Section 4.1 gives the separate complete-path timing.

E RESOURCE INTERVENTIONS, PREPARATION, AND EXTERNAL CONTROLS

For provenance, historical protocol IDs denote scientific tasks: P25 state/read exchange; P26 blind allocation; P27 compression-target control; P28 shared supervised grid; P29 high-harmonic stress; P30 state/read interventions; P31 frozen video; P32 fixed-weight iterations; P33 clipped nonlinear scope; P34 preparation cost; and P35 receiver realization.

Evidence units and source separation. P30, P32, P33 and P35 each use 24 new independent synthetic worlds, separate from P25–P29 and from one another. Their controlled receivers are selected without using final test episodes. P34 uses 12 further worlds, with three timing/training repeats nested within each world; its 432 price queries are two choices on each of 216 repeated task runs, not 432 independent worlds. Its preparation-time interval resamples worlds. P31 uses complete videos as units and does not apply the synthetic episode-level calibration guarantee to overlapping windows. The frozen protocols, outcome vectors and create-only runs are retained in the experiment records. The following give the paper's executable specifications and scope distinctions.

State/read interventions and fixed-weight iterations. P30 trains two independent affine seeds at ten physical (d, g) cells per task on F0/F1 and low/middle/high paired innovation variants; calibration and test each use 524,288 independent 32-step episodes per model. It compares $d = 1 \to 7$ at $g = 8$ and $g = 8 \to 2$ at $d = 1$, with four adjusted world-bootstrap primary contrasts. P32 trains a shared-weight iterative writer on 24 different worlds, two source structures, four fixed cells and two seeds. At one selected checkpoint, it evaluates $K = 1, 2, 4, 8$ read-time iterations using identical weights and fixed state/read budgets. The predeclared low-innovation $R(K = 1) - R(K = 8)$ means are -0.000022 on the smooth source and $+0.001151$ on the eight-harmonic source, both below the 0.005 practical scale; no fixed-budget threshold rescue occurs.

Correctly clipped nonlinear scope. P33 trains original A, final-output-clipped A, parameter-matched A and correctly output-clipped nonlinear C during both training and evaluation on 24 new F1 worlds, six physical cells and two seeds. All use separate 524,288-episode calibration and test lanes. The nine C seed-row crossings of the numerical ideal affine minimum occur in seven task/cells on five worlds; two task/cells cross in both seeds. Define $\Delta = R^*_{\text{aff}} - U_C$ and the prespecified gate slack $\Delta - 0.005$. The minimum positive *gate slack* after an 85-digit source-reference recalculation is 0.000369; hence every counted crossing has $\Delta \geq 0.005369$ to the shown precision. This is a high-precision numerical check, not a formal interval proof of the full simulator. Final clipping did not activate on the independent C test episodes, but the prescribed operation was present in the new training path. The average $R(A_{\text{outclip}}) - R(C_{\text{clip}})$ is -0.00993, with five-comparison-adjusted world interval $[-0.01180, -0.00810]$. Thus crossings establish a function-class boundary, not average superiority or an isolated computation effect.

Complete preparation-path comparison. At the primary 65,536-episode calibration size, the complete cold paths have the following means. Each path reuses its fit for both prices; the final independent research test is excluded from seconds and included in pass counts. F-then-A additionally recalibrates the newly fitted selected A on an independent lane.

Preparation path	Seconds/task	Mean selected J	Test pass
Future compiler, deploy directly	1.470	0.945	432/432
Future compiler, then independent A	5.338	0.945	432/432
Shared grid, 2,000 updates	2.125	1.698	432/432
Shared grid, 38,000 updates	26.040	0.904	432/432
Independent full A grid	29.827	0.912	432/432

Frozen-video protocol and all tested methods. DAVIS 2017 TrainVal supplies 60 training and 30 official validation videos. A fixed sequence-name hash assigns the training videos to 40 fit, 10 model-selection and 10 budget-selection units. A frozen ResNet-18 ImageNet-1K encoder maps each whole frame to a 4,096-FP32 layer-3 packet; no masks, boxes, flow, object identity or future frames reach any component. The output is the normalized xy foreground centroid. The resource proxy is $J_\lambda = d/32 + \lambda/g$; $d \in \{2, 4, 8, 16, 32\}$ and $g \in \{2, 4, 8, 16\}$ plus the legal every-frame and zero-state controls. F uses centered future-output PCA with a mean offset, rather than importing the synthetic zero-mean construction. The $q = 0.04$ outcome below is a development-adapted successor to the infeasible prespecified $q = 0.01$ study, frozen before official validation extraction. On the ten budget-selection videos, each candidate's sequence-equal empirical MSE is compared with q; among candidates at or below q, the method chooses minimum J_λ under its frozen tie rule. Overlapping windows are averaged within video. No population-risk certificate is obtained from these ten units, and the 30 official validation videos play no part in the choice.

Table 3: All tested methods on 30 held-out videos at the adapted $q = 0.04$, $\lambda = 4$ setting. Costs and MSEs list the three training seeds; "pass" means empirical video MSE below the adapted target.

Method	Selected J (seeds)	Held-out MSE (seeds)	Empirical pass
Future F	.5625, .5625, .5625	.0361, .0355, .0364	3/3
Direct-A	.5625, .625, .625	.0335, .0334, .0345	3/3
Sample-and-hold	.5625, .5625, .3125	.0284, .0302, .0337	3/3
Packet PCA	4, 4, 4	.0259, .0284, .0272	3/3
Hidden PCA	.5625, .5625, 1.0625	.0378, .0406, .0360	2/3
Linear forecast	1.0625, 1.0625, 1.0625	.0318, .0323, .0323	3/3
Shared, 8,000 updates	.75, .625, .625	.0362, .0359, .0415	2/3
Shared, long training	1.0625, 1.0625, .625	.0318, .0341, .0364	3/3

The paired video-level $R(F) - R(\text{hold})$ mean is $+0.00519$ with 95% bootstrap interval $[0.00129, 0.00941]$. The reported "pass" column is empirical MSE at the adapted threshold; it is not a population-risk certificate. Packet PCA's low risk requires the every-frame, high-cost endpoint. No head sees the mask used to score the target.

Calibration-size and timing qualifications. The 16,384/65,536/262,144/524,288 calibration prefixes are nested evaluations of fixed models. At 16,384 episodes, F delivers 216/432, short shared 266/432 and long shared or Direct-A 380/432; at 65,536 and above, all four paths evaluated at those sizes deliver 432/432. F-then-A was evaluated only at the primary 65,536 size. The separate true 16,384-episode stopping-time run reproduces the replay's choices on a fixed 24-task subset; its faster time is inseparable from F's 50% no-choice rate at that sample size. The measured CPU episode latencies have packets already resident. A separate frozen-video GPU head timing also excludes encoder and transport; neither measures network bandwidth or full-system frame inference. The primary cold paths run serially and uncontended with one PyTorch CPU thread. Their timed boundary includes data generation/loading, fit, projection when used, calibration, choice and saving. The frozen run receipt fixes source hashes and thread setting but did not capture a CPU-model or library-version manifest, so its absolute seconds are specific to the executed environment.

Cost-ordered receiver-search protocol and audit. For the two requested prices $\lambda = 1, 4$, a matched receiver search sorts all physical candidates by (J_λ, d, g), fits and calibrates them sequentially, and stops at the first passing cell. It shares every already visited physical cell across the two prices and trains F1's current-head endpoint at most once. Padded aliases are skipped. The one-seed receiver recipe, $n = 65{,}536$ calibration lane, threshold, loss range, tie rule, and 12 P34 worlds with three repeats are unchanged. Each path charges its own data generation, fit, calibration, selection and checkpoint saving; the final independent test is outside the clock. Direct future compilation is rerun in the same serial session as the search. Search runs first in repeats 0 and 2, Future first in repeat 1. This comparison was specified after the original P34 results were known, so it is a matched follow-up on exposed worlds, not fresh-world confirmation. The original complete-grid seconds and these new paired seconds are not combined into a cross-session timing ratio. The search and F take 20.009 and 1.443 seconds/task, respectively. The paired geometric search/F time ratio is 12.11, with a 95% interval $[10.12, 14.32]$ from 20,000 resamples of the 12 base worlds. The arithmetic task-ratio mean is 13.89, with world interval $[12.28, 15.39]$; repeat-specific geometric ratios are 12.09, 12.10 and 12.13. The search examines 11.81 and 14.93 candidates on average for $\lambda = 1, 4$, with a union of 16.81 physical candidates and 12.98 positive-state fits per task. All 216 choice pairs reproduce the frozen full-grid choices, so F's mean selected $J = 0.945$ remains 1.038 times the search's $J = 0.912$ (12-world interval $[1.013, 1.068]$). Selected A/E checkpoint arrays and N/projection arrays match the frozen P34 files exactly; visited calibration upper bounds agree within 10^{-6}. Only after these checks do we inherit the original independent lane-9 test result: both paths pass all 432 price queries. The follow-up manifest records an AMD Ryzen 9 9950X, Linux 6.8.0-138-generic, Python 3.12.13, NumPy 2.3.1, PyTorch 2.11.0+cu128, and one PyTorch/OMP/BLAS thread. The protocol, source hashes, complete raw outcomes and analysis are preserved in the P34 research record.

Independent receiver realization controls. P35 freezes selected cells in 24 fresh eight-harmonic F1 worlds, then fits two seeds per selected cell and fixed $(8, 16)$ challenge cell. Six arms vary 2,000/20,000 updates, width 32/128, 8,192/65,536 unique episodes, and observed/privileged conditional-mean labels. Adjusted world-mean MSE gains are 0.00771 $[0.00694, 0.00857]$ for longer training, 0.00507 $[0.00417, 0.00619]$ for short-run width, 0.000014 $[-0.000076, 0.000098]$ for extra unique data, and 0.000272 $[0.000149, 0.000484]$ for privileged labels. Width also changes parameters and inference work; these are training-realization controls, not a runtime-computation bound.

A representation map for the origin-based compiler. For a positive-gap synthetic compiled cell without final output clipping, write $p = g - 1 > 0$, let the clipped future heads be $a(m) \in [-2, 2]^p$, and let $U \in \mathbb{R}^{p \times r}$ have orthonormal columns, with $r \le d \le 8$. In exact arithmetic the choice

$$s(m) = \frac{U^\top a(m)}{2\sqrt{p}}, \qquad B = 2\sqrt{p}\, U \tag{18}$$

gives $Bs = UU^\top a$, the compiler's origin-based gap output. Every stored coordinate satisfies $|s_k| \le 1$, since $\|a\|_2 \le 2\sqrt{p}$. Each phase row obeys $\|B_{j,:}\|_1 \le 2\sqrt{pr} \le 2\sqrt{15 \cdot 8} < 32$. Unused coordinates can be padded with zero, and the receiver can reuse N's bounded writer and the same lawful read-step head. The executed P35 A0 used a different columnwise rescaling of this linear projection and separately checked all 117 cells and FP32 output parity. This construction establishes representability within the specified synthetic A interface; it does not imply that a fresh randomly initialized fit will find those weights.

Table 4: Task-specific state/read allocation at target MSE 0.04. Match compares the exact selected (d, g) with a separately validated full A grid; both independent A seeds must pass on final test. Savings are mean fractional reductions in normalized cost J_λ relative to development-chosen single-axis policies; each world first averages its three paired variants (24 independent worlds per family and price). "Fixed d" still chooses read frequency by task; "fixed g" still chooses capacity by task. All listed comparator selections also pass test. The cheap-read endpoint is discussed separately because all choices are $(0, 1)$.

Source	Price λ	Pass	Match	Save vs. fixed d	Save vs. fixed g
F0	1	72/72	72/72	10.7%	10.7%
F0	4	72/72	72/72	13.9%	34.6%
F1	1	72/72	70/72	7.6%	9.5%
F1	4	72/72	71/72	13.7%	28.7%

F Additional controlled comparisons and certificates

F.1 A concrete capacity-for-frequency exchange

In the P25 precursor, the lower-innovation world in Figure 1 has compiler bounds $U_N(2, 8) = 0.0341$, $L_{\text{aff}}^{\text{known}}(2, 16) = 0.0581$, and $U_N(4, 16) = 0.0142$ at $q = 0.04$. The two feasible endpoints pass with both independently trained A seeds. The excluded $(2, 16)$ cell is a class-level lower certificate under the P25 source assumptions, not a lower confidence limit on one trained receiver. The middle-innovation task in the same world has a supplied $g = 8$ information floor of 0.040033, so changing capacity alone cannot meet the target under that exact schedule. These are particular preselected tasks; the full P25 low-innovation group has certified exchanges in 5 of 12 independent worlds. Its 404 successful receiver checks represent 202 distinct physical configurations evaluated with two seeds, not 404 independent configurations. P25 knows the source's innovation floor. P26 asks whether resource choices remain useful when that floor is unavailable to the allocator. For this particular exchange, $J_\lambda(4, 16) - J_\lambda(2, 8) = 1/4 - \lambda/16$; it becomes cheaper only when $\lambda > 4$. Figure 1 illustrates a feasible exchange, not a strict cost improvement at the principal prices.

F.2 How well do compiled budgets match independent training?

Table 4 presents all noncheap price conditions in P26. The frozen allocator chooses 72 tasks per source family and price. Every chosen pair passes the final target. At the listed noncheap prices, both independently trained A seeds pass at every selected pair. The cheap-read F0 $(0, 1)$ choice is a deterministic feedthrough endpoint rather than a trained receiver. The selected pair matches the independent Direct-A-grid choice for every F0 query and for 141 of 144 F1 queries across the two prices. At the cheap-read price $\lambda = 0.25$, all 144 tasks choose $(d, g) = (0, 1)$, with no gain over a fixed pair and exact agreement with Direct-A. Hence the complete agreement is 429/432, and the more informative restricted readout is 285/288. The latter is a descriptive reanalysis of the frozen choices, not a new preregistered threshold. There are 348 distinct allocated task/cell combinations among the 432 queries; multiple prices can select the same cell.

The three disagreements matter despite the high match rate. F1's worst individual cost ratio to Direct-A is 1.2 at $\lambda = 1$ and 1.5 at $\lambda = 4$. A median ratio of one therefore does not establish a uniform near-optimality guarantee. Direct-A itself searches the finite trained grid, not all possible models under the same byte budget.

The already-frozen linear forecaster L is a cheaper prediction baseline under the same packet, projection, calibration and allocation rule. Its certified feasible frontier matches the analytic affine frontier at 494/648 F0 and 508/648 F1 positions, versus 637/648 and 630/648 for N; L still recovers many useful cells. At $\lambda = 4$, its chosen pair matches Direct-A in 47/72 F0 and 43/72 F1 queries, versus 72/72 and 71/72 for N. In the remaining 25 F0 and 27 F1 queries, L chooses a higher-cost pair; it never chooses a lower-cost one in these high-price queries. Both independent A seeds pass for all 144 L selections. This comparison shows when the nonlinear task forecast helps the allocation in these sources. It does not isolate future-output compression from other task-aware encoders, and L has a different fitting cost and capacity.

F.3 How much does choosing both axes add?

At high read price, joint allocation saves 13.9% in F0 and 13.7% in F1 relative to the stronger fixed-capacity policy, although that policy still adjusts g for each task. The corresponding preserved, per-comparison 95% world-bootstrap intervals are $[13.4, 14.6]\%$ and $[12.4, 15.3]\%$, respectively. A fixed-frequency policy leaves larger savings at that price, 34.6% and 28.7%. Savings against one globally fixed (d, g) are around one third at the same price, but they overstate the incremental value over either single-axis policy.

The price dependence is part of the result. At $\lambda = 1$, savings against fixed capacity are 10.7% and 7.6%. The preregistered gate that at least 12/24 worlds should save 15% against each single-axis policy fails: only 8/24 do so in either family at this price. At $\lambda = 0.25$, reading every step at zero state is the selected endpoint for both families, so there is no saving over a globally fixed pair. Thus the robust practical claim is about expensive reads and heterogeneous task frontiers in this fixed experimental family, not every price or a universal state/read law.

The two source families and innovation strengths also matter. At $\lambda = 1$, F0's savings over a fixed pair concentrate in low-innovation tasks (reported task means 25.4% low, 6.7% middle, 0% high), whereas F1's corresponding descriptive means are 25.4%, 25.0%, and 10.8%. These are paired task-level summaries, not additional independent-world confirmations. The training and calibration needed for each new task also consume work. The original P26 study trained the entire A grid to audit the recommendations, so that run did not directly save fitting time or measure hardware throughput.

F.4 Why compress future task answers?

P27 tests that narrower mechanism on 24 *new* base worlds generated from the same two source formulas with a new master seed. Before viewing their results, we fixed three compilers that share each task's trained N, packet, projection data, FP32 state count, read schedule, affine phase heads and 524288-episode calibration/test lanes. The future compiler F compresses N's unread task-head vector. Packet P instead PCA-compresses the nine current input scalars, and hidden H PCA-compresses N's current 32-dimensional ReLU features; P/H fit an origin-linear decoder from their state to N's future heads. All three use the same final $[-2, 2]$ output clip, a P27 change from the executed P26 compiler. Their executed components belong to $\mathcal{F}_{\text{clip-aff}}$, not the ideal affine class of Proposition 1; the same clip is applied to all three comparators. The six pilot worlds were used only to check implementation; the 24 confirmation worlds were not used to alter the rules.

Table 5: Future-answer versus current-feature compression: selected normalized cost J_λ (24 independent new worlds, three paired tasks per family/world). Every chosen F/P/H component passes the independent episode-level risk upper limit ($72/72$ per method, family and price). The last column is the world-mean fractional saving of F versus H with a stratified world-bootstrap 95% interval. These are compiler comparisons, not independent A transfer or a comparison with a shared supervised grid.

Source	λ	Future F	Packet P	Hidden H	Save vs. H
F0	1	0.540	0.988	0.862	36.2% $[34.6, 37.7]$
F0	4	1.380	3.064	2.802	40.5% $[38.2, 42.9]$
F1	1	0.467	0.999	0.741	35.1% $[33.3, 36.9]$
F1	4	0.905	3.845	1.733	32.5% $[30.4, 34.7]$

F has lower selected cost than P and H for every one of the 288 task/price comparisons while all three methods meet the held-out target. This is informative about actual task risk and budget choice, not just the algebraic fact that future PCA best reconstructs N's own forecast in squared error. The controls preserve packet/hidden energy and then use a task-aware decoder; they do not learn a task-supervised low-dimensional encoder. The result therefore supports future-answer compression against these specified current-feature compressors, not against every multi-configuration or end-to-end task-trained alternative. P27 uses the same finite-rank source structure as P26 and cannot resolve the source-structure limitation above.

F.5 CAN SUPERVISED TRAINING BE SHARED ACROSS THE GRID?

P28 makes the stronger method comparison on another 24 fresh worlds from the original source formulas. A single supervised model S shares a $9 \to 32 \to 16$ writer across the 19 physical positive-state cells and trains a bottleneck and affine phase head for each cell. Its 2668 parameters exceed N's 848. S-total receives 2000 gradient updates across the grid, matching N's total update count; S-per-cell receives 38000 updates, about 2000 per cell. All methods use the same fitting, selection, calibration and independent test episode lanes, the same output clip, and the same risk threshold. These P28 executed gap outputs are also in $\mathcal{F}_{\text{clip-aff}}$, so their comparison is empirical rather than a test of the affine spectral lower bound. F/P/H are rerun on these worlds; F again selects a lower-cost passing cell than P and H in all 288 task/price comparisons.

Table 6: Shared supervised grid versus future compiler: selected normalized cost on 24 paired worlds (72 tasks per family). Every listed method's selected cell passes its independent test in all 72 tasks at each price. Differences are S minus F; brackets are stratified world-bootstrap 95% intervals. S-total has the same *total* number of fitting updates as N; S-per-cell has 19 times as many.

Source	λ	F	S-total	Δ total	S-per-cell	Δ per-cell
F0	1	0.544	0.677	$+0.133\ [+0.101, +0.169]$	0.543	$-0.002\ [-0.005, 0]$
F0	4	1.385	2.068	$+0.682\ [+0.507, +0.887]$	1.382	$-0.003\ [-0.009, 0]$
F1	1	0.477	0.639	$+0.161\ [+0.140, +0.185]$	0.476	$-0.002\ [-0.005, 0]$
F1	4	0.918	1.684	$+0.766\ [+0.604, +0.957]$	0.917	$-0.002\ [-0.005, 0]$

At equal total updates, F selects a strictly cheaper cell in 41, 53, 60 and 66 of the 72 tasks in the four rows, with no S-total win. At equal per-cell exposure, the choices tie in 70–71 of 72 tasks; S-per-cell wins the remaining one or two, and F never wins. Thus P27's result is a comparison with the specified current-feature compressors, not superiority to sufficiently trained shared supervised heads. Across the 144 P28 tasks, N fitting consumes 404 CPU seconds, versus 1105 for S-total and 14479 for S-per-cell. The complete three-compiler F/P/H worker wall-time sum is 21562 seconds, versus 14476 and 25375 seconds for the respective S workers. That F/P/H worker includes two controls in addition to F, so these job totals do not establish an F-only end-to-end speedup. They show that gradient exposure and calibration work must both enter a practical comparison. The 2000-versus-38000-update contrast changes *training* work for S over the same resource grid; it does not isolate inference computation at a fixed (d, g).

F.6 WHAT SURVIVES BEYOND THE RANK-EIGHT SOURCE?

P29 keeps the packet, period grid and N/A fitting recipes but changes the target to eight harmonics of one latent phase, $S(\theta) = c\sum_{h=1}^{8} \cos(h\theta)/h$ with $c = 0.9\sqrt{2/(3\sum_{h=1}^{8} h^{-2})}$. At $g = 16$, its ideal future-mean matrix has numerical rank 14–15 in F0 and 11–15 in F1 at tolerance 10^{-10}, beyond the state cap of eight. Its spectrum has no exact rank-eight truncation. The low-innovation tasks are still ideally feasible at $(d, g) = (8, 16)$ in all 24 worlds per family; the higher-innovation variants often cannot use that long gap because of new information arriving after the read. This separates a spectral stress from a pure information-floor obstruction.

Table 7: High-harmonic receiver realization on 24 new worlds, 72 tasks per family. The analytic ideal chooses $g > 1$ in all 72 tasks at each price. N's choices were frozen across all worlds before any A fitting. The test columns count risk-upper-limit passes at N's selected cell; "both A" requires both independent seeds. The last two columns list J_N and J_{ideal} separately; the ideal reference is an unconstrained affine minimum and does not imply deployable regret where A fails.

Source	λ	N $g > 1$	N test	Both A test	J_N	J_{ideal}
F0	1	47/72	72/72	67/72	0.721	0.578
F0	4	47/72	72/72	72/72	2.363	1.432
F1	1	48/72	72/72	69/72	0.735	0.552
F1	4	48/72	72/72	66/72	2.354	1.399

All 24 low-innovation tasks in each family choose $g > 1$, despite predictive rank above eight. At the highest innovation strength, all 24 tasks per family fall back to $g = 1$ even though the ideal affine reference permits a shorter nontrivial gap. Every selected compiled N component passes test,

but independent A realization fails in five F0 tasks at $\lambda = 1$, three F1 tasks at $\lambda = 1$, and six F1 tasks at $\lambda = 4$ (with some overlap across prices). These failures are concentrated in low or middle innovation, near the risk threshold; they do not prove the selected budgets impossible. A fixed-width, fixed-training N can therefore find useful sparse-read budgets beyond an exact rank-eight cutoff, while its advice is less reliable for independently retrained receivers in this harder source.

P35 locates part of that receiver gap on 24 further high-harmonic worlds. At budgets frozen before receiver training, increasing the independent A fit from 2,000 to 20,000 updates reduces world-mean MSE by 0.00771 (five-comparison-adjusted interval $[0.00694, 0.00857]$); selected-cell seed-level upper-bound passes increase from $132/138$ to $138/138$. Requiring both seeds at a budget gives $64/69$ to $69/69$ selected task/cells. A zero-gradient mapping of the frozen compiler into the bounded receiver interface is legal at all 117 checked cells and matches its outputs to within 7.2×10^{-7} absolute FP32 error. At the 69 selected cells, the mapped model's own conservative bounds pass in 48 and are undecided in 21; none certifies failure. Failures of a fresh random A fit can therefore reflect training realization, not an impossibility of the selected budget. Longer training is not a universal repair: half the seed-level fixed $(8, 16)$ challenge cells still have certified failures in every longer or wider arm.

F.7 Feasibility, necessity, and a boundary of the component class

The frozen N predictor's certified feasible frontier agrees with the analytic affine reference at 637/648 F0 and 630/648 F1 capacity positions. Its blind upper/lower intervals meet exactly at only 207/648 and 320/648; providing the true innovation floor to an analysis-only version raises those counts to 600/648 and 564/648. The allocation choices stay the same because their upper bounds and cost rule stay the same. Incomplete brackets mean the available evidence does not *certify* minimality, not that minimality is intrinsically statistically unidentifiable. For example, at $\lambda = 4$ the selected compiler's smallest margin $q - U$ is 0.00164 in F0 and 0.00026 in F1, while the blind lower rule excludes *every* cheaper cost-and-tie-ordered cell in only 8/72 tasks per family. Hence the high agreement with Direct-A is mainly an empirical allocation result, not a blanket minimality certificate. The oracle comparisons also inherit a numerical qualification: the FP64 trigonometric source was audited, but no uniform interval proof of rounding error is available.

The original P26 nonlinear receiver C omitted its prescribed final output clip during training, so P33 retrained a correctly clipped C, an original affine A, an output-clipped A, and a parameter-matched affine A on 24 new F1 worlds. C crossed the ideal affine minimum by at least 0.005 in nine seed rows, seven task/cells and five worlds; neither output-clipped A nor parameter-matched affine A did so. The minimum gate slack after subtracting the 0.005 crossing threshold is 0.000369 (Appendix E). The classifications survived an 85-digit source-reference recomputation. Yet C was 0.00993 MSE *worse* than output-clipped A on average across its tested tasks, cells and seeds, and passed $181/864$ seed-cell tests at $q = 0.04$ versus $230/864$ for clipped A. This is evidence that the affine spectral reference cannot lower-bound every legal nonlinear gap decoder; it is not evidence that C is a better average allocator or that additional gap computation alone causes a benefit.

With weights and budgets fixed, P32's $K = 1 \to 8$ read-time iteration change stays below its prespecified 0.005 useful MSE scale and gives no K-only threshold rescue in the tested writer. Appendix E gives the full comparison and protocols.