

Cumulant Propagation under a Compute Budget

State-matched corrections, final-loss response fitting and verified arithmetic for mean ReLU activations

hyojun_kwon | ARC White-Box Estimation Challenge 2026 | revision of 18 October 2026

Corresponding successfully graded submission: #335413 (adjusted score $3.3425257575475854e-9$; raw final-layer MSE $2.4308055479593803e-8$; graded status observed 12 October 2026, KST). This is the frozen #334316 cumulant estimator plus the terminal residual corrector in Section 6. Sections 1-5 explain the underlying method and its scored lineage; Section 7 records the tested limits and later negative results. [E38]

Phase 2 nominations (reported by the participant on 18 October 2026): #336611 and #335413. #336611 is the transfer of this write-up's terminal corrector to another participant's public MIT-licensed estimator (Section 6, "Transfer to a public engine"); the engine it runs is not ours. The nominations are separate from the single submission associated with this write-up, which remains #335413. Other submission IDs identify historical comparisons.

Abstract

We estimate mean activations of width-1024, depth-16 random ReLU networks by propagating approximate cumulants. Building on Wu et al. [1], the method carries covariance and factorized third-order state, repairs selected cumulant terms, and fits corrections on the actual rank-capped float32 trajectory. It does not sample full network evaluations during prediction.

Two independently evaluated cumulant repairs each reduced adjusted error by about 4.2% on their own frozen 32-network block (Section 2). Native final-loss response fitting and verified matrix-product schedules then moved the official score from **3.798e-9** (#331357) to **3.598e-9** (#332177). Call batching allowed later arithmetic savings to grade within the residual-time limit (#333716: **3.585e-9**). A correction to the propagated covariance, restricted to layers 8-14, reached **3.541e-9** (#334316). These comparisons separate statistical changes from changes to arithmetic and client overhead; they do not assign all gains to one mechanism.

The corresponding submission adds a **6,594-parameter terminal residual corrector** to that frozen estimator. It reads 33 per-neuron features already computed by the propagation and transports two scalar messages through the weights. Training on organizer-provided public targets uses cost-weighted final-layer MSE. The three graded steps reached $3.392e-9$ (#334717), $3.373e-9$ (#334793), and **3.343e-9 (#335413)**, a **5.61%** reduction against #334316 at **0.126%** additional prediction FLOPs. This learned component is assessed separately from the mechanistic cumulant repairs. The same corrector architecture, retrained on the state of LordProcciuto's public e30_a estimator (#335981, $3.116e-9$; MIT, itself derived from 504aldo's V29) without changing that engine, graded as **#336611 = 3.044e-9**: raw final-layer MSE $\times 0.975$ and score $\times 0.977$ against the engine at $+0.14$ - 0.15% FLOPs on the two uncaptured baselines checked. The final-block raw-MSE ratio was $\times 0.978$; a later holdout scored $\times 0.976$ under its recorded cost approximation. [E48, E49] Each submitted step had development and independent final checks; a 384-network intermediate was withheld when only 37 of 64 final comparisons improved, below its registered 39-win threshold. [E37, E38]

The limits are equally specific. Small response refits showed a generalization gap; arithmetic savings could fail the 0.4 s residual-time gate even when outputs were unchanged. Matched-refit representation tests, subsequent optimizer and context variants, and a first-layer control variate did not yield a qualifying replacement. These are results for the tested implementations, data and budgets, not impossibility claims. Frozen artifact hashes, data roles, exact-versus-approximate checks, and official receipts make the scoring trail auditable. Later unsubmitted runtime candidates are explicitly separated from the graded source.

Main contributions

- A deployed-state view of calibration: fit corrections on the actual rank-capped propagation trajectory, and assess their final-layer effect rather than isolated regression fit.
- Final-loss response fitting: coefficients fitted to measured native responses of the deployed estimator, with independent optimizer checks and explicit training/development/official separation. A differentiable GPU replica was built for this purpose but failed its response-transfer gate; fits proposed through it are reported as diagnostics only, and none was promoted to a native run or a submission.
- Two explicit cumulant repairs with independent paired evaluation and a documented official scoring trail.
- A verified fast-multiplication ladder inside the metered interface: a six-level alternative-basis schedule and a selective rank-48 4×4 recursion, each with an exact bill recurrence that the measured full-estimator bill matched, component accuracy gates, and byte-level parity checks for the bill-neutral changes; an account of the measured residual-time limit; and a call-batching construction that buys residual margin back at byte-identical output.
- A reproducibility record separating numerical equivalence, metered FLOPs, API calls, transient handles, native timing, wire timing, and official grading, including two grader-interface findings (a dtype-metadata portability assertion and the local-to-official residual-time ratio).
- A learned terminal residual corrector on the frozen estimator: features captured from the deployed propagation, two transported scalar messages, the scored quantity as the training loss, organizer-provided public targets, and a fixed gate ladder (development, then an independent final block opened once) that every graded step passed and one candidate failed; with the measured limits of the lever (capacity along the hidden and the layer axes, extra state, objective, training horizon, ensembling, data: −0.9% from 256→384 networks, −0.6% from 384→760).
- Transfer of the corrector to an independently developed public estimator (e30_a): retrained on that engine's own captured state with the engine byte-identical, graded −2.3% against the engine, plus three preregistered follow-ups: the seed ensemble failed its holdout gate, the GRU stopped on development evidence before its holdout, and synthetic-data scaling failed its interim continuation rule.
- Measured negative results with their failure located, including the representation gates in Section 7, and an account from public telemetry of where the remaining gap sits.

1. Estimator and compute model

For a supplied network and standard-normal input x , the contract returns estimates of $E[\text{ReLU}(z_{l,i})]$ for all hidden layers. The score uses the final layer: $\text{MSE_final},m \times \max(0.1, F_m / B_m)$, averaged across the evaluated suite. Phase 2 uses $B = 2^{41}$ metered FLOPs, with independent wall-time and residual-time limits of 120 s and 0.4 s. Numerical work uses flopscope operations; ordinary Python organizes calls and storage. [2]

Propagated state

The starting point is the Gaussian ReLU identity $E[\text{ReLU}(Z)] = \sigma \phi(\mu/\sigma) + \mu \Phi(\mu/\sigma)$. Cumulant corrections retain information beyond this Gaussian marginal. We use the published cumulant/Hermite approach [1] as the foundation; cumulant propagation and its wide-network analysis are prior work.

The frozen source maintains covariance and factorized third-order state rather than materializing a dense third-order tensor. Its cubic contraction has the structure $\sum_s (X_s \cdot t)(P_s \cdot t)^2 + 3 \sum_s d_s (P_s \cdot t)(F_s \cdot t)^2$. Here s indexes carrier terms, X_s , P_s and F_s are their stored factor vectors, d_s is a scalar weight, and t is an arbitrary test vector. Factor directions permit tensor contractions to be implemented as matrix products and elementwise readouts. Rank truncation controls the cost of carrying this state. It also changes the approximation error on which the corrections operate.

The deployed configuration is float32, a carrier cap of 7,680 (the maximum number of factor directions carried for the third-order state), and a specialized terminal query of size 768 (padded to 832 in the multiplication schedule, Section 5). It uses shape-dependent recursive multiplication, including one deeper level of transport recursion from network layer 5 on, covariance transports, and projected slices. Four first/shared-left pairs of 1024-by-832-by-7680 transports use a six-level alternative-basis schedule; 27 cost-ranked products (the eight 832-by-7680-by-832 terminal query products, seven further products of widths 3072-6144, and the level-6 members of three compact-width families) use a rank-48 4x4 recursion (Section 5). Fitted-pair temporaries share six preallocated scratch buffers. The separate XA compiler schedule is not included in this source. The frozen source specifies the branches and coefficient tables; this description is not a replacement implementation.

Further deployed approximations include a layerwise retained-coordinate schedule, physical third-cumulant dressing of pair weights, and a two-step projection with carrier relation $F_s = \alpha_s P_s + \beta_s X_s$, where α_s and β_s are scalar projection coefficients. The retained-coordinate schedule exploits the ReLU gate: coordinates whose standardized pre-activation mean $a = \mu/\sigma$ is far below zero have Gaussian ReLU (Wick) weights that are numerically negligible, so at each layer the $n - \text{keep_l}$ coordinates with the lowest a are dropped from the transported state before the next layer's products. The keep counts are fixed per layer (1024 down to 768) and network-independent; in the release that introduced them this reduced the bill by 12-18% at a raw-MSE cost below 1%, and the correction tables are fitted on the pruned trajectory. [E19] Six rank-40 projected slice geometries and the terminal query avoid materializing unused contractions. Calibration uses local moment and row-reduction features; terminal features with no computed consumer are omitted consistently with their fitted table. These interacting choices are retained from the validated lineage. The two experiments below isolate later bundled repairs, not the individual effect of every existing design choice. [E5, E7]

Terms used below

MLP: multilayer perceptron; *MSE*: mean squared error; *MC*: Monte Carlo; *DEV*: a development set already used for selection; *CI*: confidence interval. Φ and ϕ denote the standard-normal cumulative distribution and density; μ and σ denote a neuron's pre-activation mean and standard deviation. *Carrier / factor directions*: the columns of the factorized third-order state; *cap*: their maximum number per layer. *K3, K4, K22*: the learned per-layer coefficient channels acting on the third-cumulant diagonal, the fourth-cumulant diagonal, and the four pair-term coefficients. *P11, P21, P22*: the pair-moment blocks consumed by the next layer's Wick weights. *Native*: an in-process run of the estimator with the metering library; *client/server (wire)*: the same estimator run through the public client/server harness that mirrors the grader's execution path; *handles*: remote array objects, capped at 800 by that harness. *Residual time*: wall time inside `predict` not attributed to metered operations. *Adjusted ratio*: candidate-to-baseline ratio of $MSE \times \max(0.1, F/B)$ averaged over paired networks. *C47* denotes two rectangular level-6 products with retained views; *B60* denotes nested rank-48 query products plus classical left-tree reuse and runtime changes; *SFP* denotes symmetric fitted-pair work; *ITN* an identity-norm cut; *HFP* a terminal quadratic variance-readout schedule; *HBR* adds copy batching to *HFP*. These are experiment labels, as are *P2VP*, *XA*, *XD* and *basis832*, rather than additional mathematical objects. *CP* denotes a sum of rank-one tensor factors. *GELU* is the Gaussian error linear unit; *GRU* is a gated recurrent unit.

Why match the calibration state?

Earlier diagnostics on a different, untruncated configuration used three MLPs and 300-million-sample references. They found very accurate carried mean, variance and third-cumulant diagonals, but much weaker fourth-cumulant diagonal correlation (approximately 0.178). This motivated local moment-feature corrections. It was a diagnostic of that configuration, not an ablation of the current incumbent. [E7]

Rank truncation subsequently changed the usefulness of third-cumulant corrections. A correction harmful on one trajectory could help on a more strongly truncated trajectory: with the same channel, features and fitting procedure, the learned third-cumulant channel worsened the development ratio on the untruncated trajectory (0.739 against 0.729 without it) but improved it on the deployed capped trajectory (0.712 against 0.953), and a table fitted at one cap was 24% worse when applied unchanged at a lower cap. [E19] Fitting tables off the deployed trajectory was therefore an implementation error, not a valid negative result about learned corrections (Section 3 applies the corrected procedure). The final method fits corrections to the state it propagates. This supports a practical design rule, not a universal theorem separating learnable and unlearnable errors.

2. Two measured cumulant repairs

Mixed covariance/third-cumulant pair term

Let C_{ij} be off-diagonal covariance, $T_{ij} = \text{cumulant}(z_i, z_i, z_j)$, and $W_i(a, p)$ the a -th mean derivative of the Gaussian expectation $E[\text{ReLU}(z_i)^p]$. Expanding the cross-neuron cumulant operator produces the mixed increment

$$\Delta P_{pq} = (C_{ij} T_{ij} / 2) W_i(3, p) W_j(2, q) + (C_{ij} T_{ji} / 2) W_i(2, p) W_j(3, q).$$

The implementation applies this increment to P11, P21 and P22 with the corresponding internal scaling. Four work buffers are reused. The Gaussian input has no initial third cumulant, and the final layer has no downstream pair consumer, leaving 14 effective layers. The extra bill was 220,200,960 FLOPs. A reference check preceded evaluation of the fixed float32 implementation.

An exploratory development extension followed an initial pilot stop; it is not presented as a pristine preregistered success. The subsequent frozen independent block B (800-831) gave adjusted-score ratio **0.9579537794**, 32/32 wins, and paired 95% interval [**0.9513659585**, **0.9646641419**]. Historical official submissions moved from #330028 at 4.2547770512e-9 to #330091 at 4.0763488807e-9. [E2]

Covariance-shaped fourth-order template

For covariance C , define $\text{Gamma}(C)_{ijkl} = C_{ij} C_{kl} + C_{ik} C_{jl} + C_{il} C_{jk}$. Given the modeled double trace S , use $K4_tilde = S \text{Gamma}(C) / ((\text{tr } C)^2 + 2 \text{tr}(C^2))$.

$\text{Gamma}(C)$ is the fourth-moment pattern of a Gaussian vector with covariance C , so the template inherits the covariance's anisotropy while preserving the modeled double trace; a scalar isotropic approximation has no such shape. It is generally oblique, not a nearest-tensor or contractive projection. The implementation changes raw fourth moments and raw P22, including the previously missing $2 U_{ij}^2$ contribution. Consequently this is a bundled repair, not a pure ablation of shape alone. It uses no new fitted parameters and reuses covariance transport.

Twelve dense reference checks agreed to approximately 1.02e-15. Frozen independent block D (864-895) gave ratio **0.9581261104**, 31/32 wins, and paired 95% interval [**0.9484229166**, **0.9677322760**]. Historical official #330159 scored **3.8828317480e-9**, with raw MSE 2.6991861013e-8. Later arithmetic versions form the path to the corresponding submission. [E2]

3. Matching calibration to the literal implementation

The current K4 table was refitted on the actual float32 trajectory, using networks 40-79 for training and exposed development networks 80-87 for comparison. The three-round endpoint and ridge coefficient $1e-7$ were fixed in advance; the K3 table, rank policy and dressing gains stayed fixed. Each new round collected a new trajectory with the preceding table. A local fixed-feature least-squares residual was not represented as the final rolled-out training loss. [E8]

This distinction mattered. A float64 replica could approximate the overall output while predicting the effect of a small parameter change poorly: the measured differential mismatch was 42.8-48.3%, above its 5% gate. We therefore used a read-only feature observer on the deployed source, first verifying that adding it preserved all output bytes and metered prediction FLOPs. Candidate tables were subsequently ported to an uninstrumented estimator and checked again. [E8]

A retained implementation discrepancy

The source-form fifth Wick coefficient used in dressing scales as σ^5 , whereas the Gaussian derivative identity requires σ^4 . Both forms were evaluated with separate matched K4 refits: the corrected form won on the eight development networks, but the retained-form submission #331350 ($3.8029e-9$) graded 0.03% better than the corrected-form #331337 ($3.8041e-9$). The retained form and its matched table are kept as an implementation fact, not as a mathematical claim: the two tables differ, and the local ranking did not transfer exactly. [E8]

Final nonnegative projection

The final output applies $\max(\text{estimate}, 0)$. Since a true ReLU mean is nonnegative, this coordinatewise projection cannot increase squared error against that target. Replacing the previous copy with this operation had the same measured FLOP bill. Official #331353 improved #331350 by 0.0004854%, with 100 normal cases. This is a small verified implementation improvement, not a new cumulant-propagation result. [E9]

4. Fitting corrections to the final loss

Section 3 matched the *state* on which corrections are fitted. The corrections themselves were still fitted to moment targets (for example the fourth-cumulant diagonal). Two objectives are in play: matching the propagated moments to reference values, and matching the final-layer mean that the score pays for. On the truncated trajectory these are not the same objective, because the readout is insensitive to some carried-moment errors and sensitive to others, so fitting to moment targets leaves score that fitting to the final loss recovers. From 19 September we therefore fitted a small coefficient vector θ directly to pooled final-layer MSE of the literal float32 estimator.

Measured responses, not a surrogate gradient

For each training network, the deployed source is run at $\theta = 0$ and at ± 1 endpoint per coordinate, each in a fresh process with one BLAS thread, and the final-layer predictions are recorded. Central differences give a response matrix J (step sizes fixed in advance: 0.05 for K3 coordinates, 0.01 for K4, 0.25 of the coefficient scale for pair terms). With the target $GT - \text{parent prediction}$ stacked over networks, where GT is the dataset's published per-layer reference mean (`all_layer_means`, 1e9 input samples per network), we solve a bounded, column-normalized ridge least-squares problem on pooled final MSE. The parameter box, step sizes and ridge grid are frozen before collection in the preregistered protocol [E12]. Every fit is re-solved independently (L-BFGS-B, relative objective agreement $\leq 1e-7$, projected-gradient check), and $\theta = 0$ is verified to reproduce the parent's output bytes and FLOP bill exactly. The linear model is only a proposal: each candidate is re-run natively on training and development networks, and selection uses those native results.

Responses were measured on 42 coordinates: relative K3 gains on layers 1-14, additive changes to the K4 variance-squared term on layers 1-14, and one common relative gain for the four pair (K22) coefficients per layer; the first and last layers are fixed. The accepted fits (#331910, #332017 and hence the corresponding submission) use the 28 K3 and K4 coordinates, with the K22 gains held at their #331604 values.

A differentiable float32 GPU replica reproduced the forward pass (parity $< 1e-5$) and would have allowed fitting on 200 networks. It failed its pre-registered transfer gate: for a fixed K4 layer-1 step, its predicted score ratio differed from the native one by 0.67 percentage points against a 0.3 point limit. It was therefore not used for any accepted fit. On this evidence, fine-scale responses of this estimator are a property of the float32 code path, not only of the mathematics.

Path of official results

Submission	Change	Official score	vs previous row
#331548	arithmetic-only parent	3.7487150148e-9	—
#331604	three staged K22 gains (0.5/1.5/0.5), native final-loss fit	3.7032655062e-9	−1.212%
#331650	K4 final-loss fit (exploratory; local development gate narrowly failed)	3.6802742328e-9	−0.621%
#331740	joint K3/K4 fit, networks full0-7, relative ridge 0.5	3.6712699581e-9	−0.245%
#331836	42-coordinate layerwise fit, 16 new networks (full32-47), ridge 5	3.6617000040e-9	−0.261%
#331910	same responses, K3+K4 coordinates only, ridge 0.5	3.6551101518e-9	−0.180%
#332017	same coordinates, 32 training networks (full32-63, 2,720 responses), ridge 0.5	3.6472948196e-9	−0.214%

Rows after #331604 include exploratory submissions made on the same public networks; several sibling configurations were also graded and are reported in Section 7. The gains shrink overall, from about 1.2% to about 0.2% per accepted step. #332017 is the last accepted accuracy step of this response-refit series; #332177 and its arithmetic descendants through #333716 inherit its coefficient tables unchanged and differ only in arithmetic and runtime organization (Section 5).

What limited it

In the measured response fits, independent optimizer checks passed while the development results showed a generalization gap. In the 16-network layerwise fit, ridge 0.05 improved training MSE by 1.37% yet worsened development MSE by 0.17%; ridge 5 gave −0.34% and −0.18%. The linear response model also over-predicts: for ridge 0.5 it predicted −1.66% on training networks against −1.25% measured natively. Internal cross-validation within the training set improved from 4 to 8 networks but was flat from 8 to 12. Doubling to 32 training networks (2,720 native responses) at a fixed ridge of 0.5 improved development MSE by a further 0.30% relative to the same-ridge 16-network fit, and by 0.29% relative to #331910 (6/8 wins, worst +0.90%). Officially, #332017 improved #331910 by 0.214% (33 wins, 1 tie, 16 losses on the 50 public networks; worst network +1.04%; all-layer MSE −0.29%). Development networks were used for selection, so the local figures are not independent test results.

Scaling further did not transfer. Fits proposed on 200 training networks through the GPU replica (three ridge values, three epochs each) improved replica training MSE by 0.9-1.1% but worsened replica development MSE by 0.25-0.47% (2/8 wins), and a training-size diagnostic at 8/16/32/64/128 networks met the

preregistered -2.5% development gate at no size; none reached the native stage. A native relinearization at the current coefficients (672 central responses on eight further networks) improved training by 1.3% and worsened development by 0.95% (2/8 wins). Two zero-cost diagnostics on the 32 training networks, a shared final-output offset and a shared multiplicative gain, had oracle improvements of 0.010% and 0.004% and worsened leave-one-out error. On the native evidence available (16 to 32 networks, the offset and gain diagnostics), the remaining error of this estimator does not appear to be a residual removable by the tested shared offset, gain, or response-refit variants; the replica scaling result is suggestive only, because the replica failed its transfer gate.

Final-loss fitting can trade intermediate-layer accuracy for the scored layer: for example, #331910's all-layer MSE on the 50 public networks with exposed per-network results is 2.96% worse than #331836's. The contest scores only the final layer, but we do not describe these refits as improving the estimator's intermediate-layer accuracy.

5. Arithmetic and official evidence

The tensor representation makes large matrix products the principal arithmetic target. Shape-dependent recursion reduces multiplication work; preallocated storage and reused views reduce bookkeeping overhead. The P2VP shared-left change removes redundant preparation for operands with shared linear structure. The 17 September release (#331357) replaces one repeated transport geometry with a packed alternative-basis multiplication schedule. These changes do not change the statistical objective, but their float32 roundoff can change the predictions.

Three further arithmetic changes on 18-19 September are inherited by the corresponding source, each checked in the same way: 64-element minimum padding with left-operand sharing for the 832-wide transports (#331480), a variable-width compact basis (#331538), and an extension of the multiplication schedule to additional operand shapes (#331533, #331548). Together they lowered the official score from #331357's value by 1.29% (#331548). The padding/sharing change accounted for most of this (#331480, -1.21%); the later two moved the score by less than 0.1% each, which is why the work then turned to accuracy (Section 4). [E15]

Exact algebra alone is insufficient in float32. We separately check output bytes or stated tolerance, FLOP accounting, repeated use of an estimator instance, storage and handles, and the execution path. The corresponding source and submitted tarball are frozen by SHA-256. No later compiler candidate is substituted for them.

A restricted alternative-basis implementation

The bilinear identity is prior work: Schwartz and Vaknin [3, Eq. 3.1] express matrix multiplication using seven products in an alternative basis. We implement that identity through the allowed flopscope interface and restrict it to shape (1024,832,7680) at six recursive levels. Every other multiplication geometry remains the parent implementation. There is no new fitted table or statistical correction in this change.

Each root copies its operands into preallocated buffers, applies basis transforms, recurses while retaining the transformed representation, decodes the result, and copies it to the original output. Consecutive paired products reuse the transformed left operand and its recursive preparations. All transform arithmetic and copies inside prediction are charged; only storage and reshape views are prepared in setup.

At a nonleaf level, each operand has four quadrant copies and three additions; recombination uses six additions/subtractions. The leaf forms three derived quadrants per operand and executes seven matrix products. The first and shared-left root bills are 6,493,204,584 and 6,448,669,080 FLOPs. Four pairs save exactly **335,364,096 FLOPs** against the parent, including input transforms and output restoration. The observed full-estimator bill matches this recurrence. [E10]

Small multilevel integer cases were exact. Two full-size synthetic products had relative Frobenius errors 2.6874e-5 and 2.6698e-5 against float64 multiplication. Exposed development 80-87 gave adjusted ratio 0.9984688654 with 6/8 wins; repeated-instance and two fallback checks passed. These are numerical and development checks, not fresh independent evidence. A local wire observation matched native bytes/FLOPs and used 788/800 handles, but its 0.5639523 s residual failed the local time cap. That failure is preserved alongside the subsequent official success. [E10]

Selective rank-48 recursion and pooled storage (24 September)

The second arithmetic round replaced two-level Strassen recursion (49 products per 4×4 block) at selected products with the regular canonical rank-48 non-commutative 4×4 scheme of Dumas, Pernet, Sedoglavic and Tichavský [4] (48 multiplications and 216 additions or scalings per block). The bilinear coefficients were frozen from the authors' PLinOpt repository (commit [6014d51](#)), parsed by a mathematical-AST whitelist rather than executed, and checked for exact rational agreement before any float work. The scheme is applied as one extra recursion level on top of the existing Strassen ladder and only where a fixed, network-independent predicate selects it. The predicate ranks the eligible products of the reference topology by metered bill saved under the rank-48 scheme and keeps the 27 with the largest savings, a cut fixed before any network-level test: the eight 832-by-7680-by-832 terminal query products, seven further products of widths 3072-6144, and the level-6 members of three compact-width families (23 fresh and 4 reused products). Two earlier rank-48 variants (the 10 September canonical maps and a sparse variant) failed their component accuracy gate (per-repetition error 1.10 - $1.18 \times$ the parent's) and were rejected before any network-level test; the accurate rational variant with direct float32 output passed it (error $\leq 1e-4$ and $\leq 0.5 \times$ the parent on all 105 component calls). Applying rank-48 to every eligible product (D16/C22) improved development adjusted score by 1.09% (8/8 wins) but failed the residual-time gate described below; the 27-product selection retains most of the bill reduction with 3,516 instead of 6,644 incremental arithmetic calls. [E16]

Three further changes are inherited by the corresponding source, each preregistered with an exact bill prediction that the measured full-estimator bill then matched: (i) six preallocated scratch buffers replace per-call temporaries of the fitted-pair matrices (bill-neutral), and seven ordinary 1024-by-1024-by- n products ($n = 1024$ -4096) are promoted from recursion level 5 to 6, saving 1,344,014,336 FLOPs; (ii) for four pruned transport pairs whose compact operand is a prefix of the wide one, the wide product is computed first and its transformed left operand reused, skipping one level-6 left tree each ($1659 \cdot n \cdot k / 32$ FLOPs per tree; 54,362,112 FLOPs for the last pair); (iii) contiguous scratch views for the fixed query geometry, which change no bytes and no bill but remove 24 slicing calls. The predictions of the #332177 source differ from #332017 only by float32 regrouping (raw MSE +0.05%); its median bill is 1.39% lower. [E16]

One grader-interface finding preceded these submissions. The first rank-48 submission (#332149) failed all 100 networks with ESTIMATOR_EXCEPTION after passing the smoke test. A diagnostic on the public client/server harness reproduced it as a metadata assertion: on that path the array dtype attribute is a string, so `A.dtype == fnp.float32` is false although the arrays are float32; native NumPy accepts the scalar type. The assertion, not the arithmetic, was changed (#332154). Such checks should compare against the canonical dtype string. [E16]

The residual-time gate is the binding constraint

Every level of fast-multiplication recursion trades metered FLOPs for unmetered client interactions: calls, view constructions and allocations. Under the Phase 2 rules those interactions carry no price, but any network whose residual time exceeds 0.4 s is a failed evaluation, scored as a zero prediction with no discount [2]; #332159 below failed on all 100 networks and received no score. The fast-multiplication ladder therefore has a hard end that the FLOP bill alone does not reveal, and the campaign below locates it for this estimator. At this operating point the #332177 source uses 0.35-0.37 s per network. Of 44 further preregistered arithmetic candidates (C21-C64, 24-25 September), more than twenty passed the development gate with adjusted ratios of 0.984-0.991 against #332017 (8/8 wins each; the best, nested rank-48 at the eight query products combined with classical left-tree reuse, reached 0.9839). Apart from the four submitted, every one of them raised the forecast official residual to 0.381-0.426 s, above the 0.38 s planning margin we set against the gate, and was not submitted. In our paired client/server measurements, the changes that reduce the metered bill further (more rank-48 products, deeper recursion, nested schemes) were exactly the ones that moved the residual, through added calls, views and allocations. The remaining candidates in that September campaign were runtime-only restructurings (call batching, view caching, dedicated caches, garbage-collection scoping, storage layouts); none recovered the 10-30 ms needed there. The successful October batching construction below was a later result. [E16]

Two measurement rules came out of this. First, absolute local residuals do not transfer: the same source measured 0.39-0.45 s locally against 0.319 s officially, so the campaign forecast residuals as a paired ratio against a graded anchor measured in alternating runs on the client/server path. Even so, the first graded candidate's official maximum residual exceeded its forecast by 8.3% (0.357 s forecast, 0.387 s official), and a candidate forecast from native timing at 0.363 s failed officially at 0.434 s (#332159); later forecasts used only the client/server path and a 0.38 s planning limit. Second, a forecast inside the margin is still a forecast: of the four campaign submissions (in addition to the pre-campaign #332149), three graded (#332154 at 3.6189e-9, #332177 at 3.5984e-9, #332191 at 3.5991e-9) and one failed on the residual gate. Two arithmetic-only siblings graded on 25 September after this analysis was frozen (#332286, which adds two rectangular level-6 products to #332177, and #332292, a runtime-only variant with bit-identical output and bill) are included in the lineage table for completeness; they change no conclusion.

Official grading and execution

Official receipt, September 24	#332177 (operating point of Sections 1-5)
Adjusted score	3.5984041802143646e-9
Raw final-layer MSE	2.6123130929534e-8
Reported FLOPs (min / median / max)	302,632,267,961 / 302,938,458,641 / 303,249,335,218
Normal / failed MLPs	100 / 0
Wall time (median / max)	101.775 s / 105.494 s of 120 s
Residual time (median / max)	0.362 s / 0.369008081 s of 0.4 s
Versus previous best (#332017)	1.340463% lower adjusted score

The official lineage since the 17 September release, evaluated using a 100-network grading suite (per-network scores exposed for 50). For rows through #334316, the harvested telemetry records 100 normal completions unless failures are stated. The #333546-#333716 rows were graded on 2-3 October; #333634, #333685 and #333716 have byte-identical predictions to #333561. The failures of #333552, #333553 and #333561 are on sealed networks, so their public scores do not reflect them. The three corrector rows have confirmed GRADED status and public aggregate scores, but their complete per-network runtime/failure telemetry was not harvested. Dates below use UTC for the September/early-October receipts (the #334316 status was observed on 6 October UTC) and explicitly labeled KST for the corrector records; the latter are status-observation dates, not exact grade-completion timestamps:

Submission	Change	Adjusted score	Raw MSE	Median FLOPs	Max wall / residual (s)
#331357	17 Sep release (Sections 1-3)	3.7977635269e-9	2.6875e-8	310,751,473,909	101.18 / 0.286
#331548	padding, sharing, compact basis	3.7487150148e-9	2.6874e-8	306,772,711,181	101.44 / 0.315
#331910	final-loss refit, 16 networks	3.6551101518e-9	2.6166e-8	307,213,281,092	97.96 / 0.306
#332017	final-loss refit, 32 networks	3.6472948196e-9	2.6110e-8	307,213,279,873	105.86 / 0.319
#332154	selective rank-48, dtype fix	3.6189219940e-9	2.6120e-8	304,706,015,761	108.43 / 0.387
#332177	+ pooling, early recursion, prefix reuse, query views (operating point of Sections 1-5)	3.5984041802e-9	2.6123e-8	302,938,458,641	105.49 / 0.369
#332191	#332177 with classical left-tree reuse at three products	3.5990751918e-9	2.6122e-8	303,013,122,577	103.04 / 0.359
#332286	#332177 plus two rectangular level-6 products (graded 25 Sep)	3.5957339446e-9	2.6111e-8	302,858,039,825	106.94 / 0.375
#332292	#332286 runtime-only variant, bit-identical output and bill	3.5957339446e-9	2.6111e-8	302,858,039,825	113.85 / 0.384
#333031	held candidate C60 plus four runtime-only changes (29 Sep)	3.5897345234e-9	2.6116e-8	302,287,274,001	107.73 / 0.381
#333203	#333031 with symmetric pair work mirrored, byte-identical output (30 Sep)	3.5886884514e-9	2.6116e-8	302,199,193,617	109.10 / 0.386
#333546	+ identity-norm cut (ITN)	3.5884917596e-9	2.6116e-8	302,182,631,953	104.35 / 0.387
#333552	#333203 + terminal readout (HFP); 4 failed	3.5849793345e-9	2.6116e-8	301,887,995,921	118.70 / 0.440

Submission	Change	Adjusted score	Raw MSE	Median FLOPs	Max wall / residual (s)
#333553	HFP + copy batching (HBR); 3 failed	3.5851287728e-9	2.6116e-8	301,900,578,833	115.04 / 0.406
#333561	HFP + ITN; 8 failed	3.5847826434e-9	2.6116e-8	301,871,434,257	112.22 / 0.414
#333634	+ batched slice pairs (3 shapes)	3.5847826434e-9	2.6116e-8	301,871,434,257	105.12 / 0.369
#333685	+ batched slice pairs (6 shapes)	3.5847826434e-9	2.6116e-8	301,871,434,257	110.82 / 0.384
#333716	+ batched query pair	3.5847826434e-9	2.6116e-8	301,871,434,257	106.52 / 0.354
#334316	+ layers 8-14 covariance correction and copy batching (frozen base of Section 6; 6 Oct)	3.5410861718e-9	2.5784e-8	302,029,459,449	106.59 / 0.347
#334717	+ learned terminal residual corrector, pretrained 600 epochs on 256 networks (8 Oct KST observed)	3.3918951472e-9	2.4667e-8	+380,720,068 per network	not harvested
#334793	corrector continued with the terminal objective, 256 networks (8 Oct KST observed)	3.3726626408e-9	2.4527e-8	same	not harvested
#335413	corrector retrained on 760 networks (corresponding, 12 Oct KST observed)	3.3425257575e-9	2.4308e-8	same	not harvested

#333031 re-entered the held campaign candidate C60 (nested rank-48 at the eight query products plus classical left-tree reuse; development ratio 0.9839 against #332017) with four runtime-only changes at bit-identical component output; on eight freshly drawn synthetic networks its adjusted ratio against the #332286 source was 0.9970 (7/8 wins). #333203 computes the fitted-pair pointwise work on one symmetric region and mirrors it: outputs are byte-identical and every network's bill is 88,080,384 FLOPs lower. The residual gate stays binding: the official maxima leave 19 ms (#333031) and 14 ms (#333203) of residual margin, which is why neither is among our nomination choices. [E26] The 2-3 October rows are explained under "Client calls, not FLOPs, bind the last 0.1%" below. [E31, E32]

The September 17 arithmetic release that Sections 1-3 and this section describe in detail is kept below for reference.

Official receipt, September 17	Historical release #331357
Adjusted score	3.7977635269388284e-9
Raw final-layer MSE	2.6874756891004382e-8

Official receipt, September 17	Historical release #331357
Reported full-MLP FLOPs	310,751,473,909
Normal / failed MLPs	100 / 0
Maximum wall time	101.179024968 s / 120 s
Maximum residual time	0.286449225 s / 0.4 s
Direct parent comparison	0.1352674% lower adjusted score than #331353

The source/archive digests, 100 unique completed evaluations, constant FLOP bill and score/raw cost multiplier were checked. Raw MSE fell by 0.0274930% relative to #331353, alongside the 0.107804% bill reduction. The score also improved on the separate XA submission #331354 by 0.0695427%. Different grader runs do not isolate hardware variation or prove a causal runtime speedup. These are public results, not a private-evaluation guarantee or a rank claim. [E9, E10]

Calls are not FLOPs

A call-reducing compiler variant (XD) cut arithmetic calls by 6.1% (22,467 to 21,095) with unchanged output bytes and FLOPs, but static subscript requests rose by 2,226 and the measured residual did not fall: fewer arithmetic calls did not mean fewer client interactions, and an unchanged bill offered no adjusted-score gain. Its one official grade (#331355) matched its parent's score exactly. The variant is not part of the corresponding submission. [E4, E9]

Client calls, not FLOPs, bind the last 0.1% (2-3 October)

After #333203 the remaining arithmetic candidates were bill cuts of about 0.1% or less. ITN skips a norm computation over an exactly known identity region (16,561,664 FLOPs per network, byte-identical output). HFP computes the last layer's variance readout $\text{diag}(W C W^T)$ by a fixed terminal quadratic schedule (311,197,696 FLOPs per network below #333203; raw MSE $-3.7e-6$ relative, from float32 regrouping). It lowered the public score by 0.103%, yet failed the residual gate on sealed networks (Section 7). Across all our graded runs, once a run's median residual exceeds about 0.37 s some networks cross 0.4 s, and nearly identical runs drift by about ± 10 ms in median (#333552 and #333561 differ only by ITN, which lowered the residual on its own, yet their medians differ by +10 ms). [E31]

The residual is dominated by per-call client overhead, not by arithmetic. A local census of the metered operation log (seed 0) counts 28,018 client operations per prediction for #333203, 28,058 for ITN and 28,569 for HFP+ITN; the +551 operations are consistent with the paired official shift of +5.0 ms in median against #333203. The two same-shape products of the (2,1) slice, $\text{out} = U P^T + V X^T$, account for 10,116 of the 28,569 operations (35%), each running its own multilevel recursion. [E31, E32]

Batching construction. P and X are adjacent planes of one carrier bank (or of a stacked gather buffer), and U and V are written into adjacent planes of one stacked scratch, so the pair runs as one call batched over a leading axis of length 2 with the same per-plane arithmetic; one addition combines the (2,n,n) output. Outputs are byte-identical, the bill is unchanged and the recursion's client calls are halved. Three constraints shaped it. (i) The 800-array limit of the client/server harness counts the preallocated views carved in setup, not only temporaries: the pre-batching build peaked at 788 live arrays, and 50 additional dummy views reached the limit and aborted the run. Batched views therefore replace the two-dimensional ones instead of

being added (peak 794). (ii) The scratch arena is a fixed 3.9 GB float32 buffer. For the three 896-wide rank-40 geometries and for the 832-by-7680-by-832 rank-48 query pair a fully batched recursion does not fit (1.19e9 floats against 9.76e8 for the query pair), so only the linear-combination stages are batched over the two planes and the child product runs per plane. (iii) The rank-48 kernel is generated straight-line code; its subscripts were rewritten with a leading ellipsis, which is the identity for the existing two-dimensional roots, so one program serves both. Two generator scripts apply these rewrites to the frozen parent source and assert every substitution count. Unlike XD above, which added subscript requests, batching lowers the total count of logged operations, and the official residual follows it. [E32]

Submission	Batched pairs	Operations per prediction	Residual median / max (s)	Paired median vs row above	Failed
#333561	none	28,569	0.384 / 0.414	—	8
#333634	three rank-40 geometries	26,825	0.364 / 0.369	−19.6 ms (lower on 99/100)	0
#333685	all six rank-40 geometries	25,755	0.358 / 0.384	−5.1 ms (72/100)	0
#333716	+ rank-48 query pair	24,435	0.339 / 0.354	−19.0 ms (100/100)	0

The four rows have identical predictions and bills; paired differences match per-network official residuals by network name. The implied rate, about 5–14 μ s per removed operation, includes run-to-run drift; #333685's wider tail (within-run standard deviation 6.8 ms against 2.7 ms for #333634) shows that single-run maxima do not rank close builds, while medians and paired differences do. Against #332191, the earlier first nomination choice, #333716 is 0.40% lower in adjusted score and 12.6 ms lower in paired median residual (lower on 95/100), with a wall maximum 3.5 s higher (106.5 s against 103.0 s). Its 45 ms residual margin is one graded run, not a distribution-wide guarantee. [E31, E32]

Official receipt	#333716 (October 3)	#334316 (October 6; frozen base of the corresponding #335413)
Adjusted score	3.584782643388719e-9	3.5410861717706102e-9
Raw final-layer MSE	2.6116237847872982e-8	2.578448754064766e-8
Reported FLOPs (min / median / max)	301,565,243,577 / 301,871,434,257 / 302,182,310,834	301,723,266,043 / 302,029,459,449 / 302,283,232,960
Normal / failed MLPs	100 / 0	100 / 0
Wall time (median / max)	99.620 s / 106.525 s of 120 s	101.603 s / 106.590 s of 120 s
Residual time (median / max)	0.339 s / 0.354454667 s of 0.4 s	0.335 s / 0.346804512 s of 0.4 s
Versus earlier entries	0.379% / 0.109% lower than #332177 / #333203	1.219% lower than #333716

The bill levers examined in that campaign did not produce a further qualified score improvement. Query nesting (C48) and the early-root promotions are already in the line; C44 (full rank-48 tiling with a seven-tree cache) was forecast at 119.2 s wall and 0.426 s residual on 25 September; Strassen quadrant copies are 1.66% of the #333716 bill and structural. The bill-neutral operation cuts in this comparison buy residual margin without changing the score. [E16, E32]

6. A learned terminal residual corrector on the frozen estimator

Why a residual, and why only at the readout

By 6 October, the tested response refits and representation variants had not supplied a further qualified replacement for #334316; the Gate H covariance correction was already included in that baseline (Section 7). What remained unexplored was a different object: the estimator computes, for every layer and neuron, a set of scalar quantities (pre-activation mean, variance and standard deviation, the standardized mean and the Gaussian pass probability and density, the two Gaussian ReLU moments, the raw and fitted third- and fourth-cumulant diagonals, the row energies of the weight, of the off-diagonal covariance and of the (2,1) third-order slice, the three Wick readout coefficients, the baseline prediction and the layer index) that are not used jointly anywhere. A learned map from these quantities to the error of the scored output does not touch the propagation and cannot destabilize it, unlike the earlier tested learned closures that fed predictions back into the state (Section 7 and Phase 1). We therefore froze the #334316 estimator byte for byte and fitted a residual corrector on top of it.

Model and cost

Thirty-three features per neuron and layer are captured into unused rows of an existing metric array while the frozen estimator runs, so no additional arrays are allocated against the 800-array limit. After the estimator returns, a separate tail runs a 16-layer free rollout: at layer l the input is the standardized feature vector (train-only mean and standard deviation per feature and layer, clipped to ± 10) concatenated with two transported scalar messages, $\Phi_l \cdot (u_{l-1} \cdot W_l)$ (entrywise product with the ReLU pass probability) and $q_{l-1} (W_l \cdot W_l)$ (carried by the entrywise-squared weights); a two-hidden-layer network (GELU, $35 \rightarrow 64 \rightarrow 64 \rightarrow 2$) emits the next messages u_l, q_l . The final prediction is $\max(0, b_{16} + s \cdot u_{16})$ with b_{16} the frozen float64 baseline and s a fixed target scale; layers 1-15 are returned unchanged. The corrector has 6,594 parameters embedded in the submission source. Its metered cost is 380,720,068 FLOPs per network, 0.126% of the bill; its measured wall-time cost is about 0.3% (Section 6, "Timing").

Data, objective and gates

Targets are the organizer-provided mean activations (10^9 samples per network) of public full-split networks; features and the baseline come from one run of the frozen estimator per network (about 70 s each, 3 workers). Pretraining used 256 networks (full 608-863) with an auxiliary all-layer residual objective, batch 8, AdamW (5e-4, weight decay 1e-4, clip 1.0), 600 epochs, deterministic float32, and model selection by the cost-adjusted development score on full 864-927; the graded result is #334717. All later steps start from that pretrained state and use the *terminal* objective, which is the scored quantity itself: $\text{loss} = \text{mean}_i \alpha_i \cdot \text{mean}_n ((\max(0, b_i + s \cdot u_i) - t_i)/s)^2$, with $\alpha_i = \max(0.1, (F_i + 380,720,068 + 32,768)/2^{41})/\bar{c}$ the per-network official cost factor normalized by its frozen 256-network mean (so that adding networks does not change the gradient scale). Training then differs only in the pool: +28,800 updates on the same 256 networks (selection at update 1,152) gave #334793; +5,760 updates on 384 networks (full 32-159 added; selection at 2,592) gave the candidate 055d; +11,400 updates on 760 networks (full 0-15, 160-383, 864-999 added; 120 data epochs; selection at 9,405) gave #335413. After pretraining, development selection used full 384-447; the independent final blocks were full 256-319 (#334717), 320-383 (#334793), 544-607 (055d) and 448-463 and 480-527 (#335413), each opened once for one candidate; full 16-31 and 528-543 stayed protected and 464-479 was excluded because it overlaps the public grading networks. For each candidate gate, its training data, development set and final set are disjoint by network identity and weight digest; some retired evaluation networks enter later training pools. No official score was used for fitting or selection.

The terminal-objective continuation and expanded-pool candidates used ratio ≤ 0.995 , paired bootstrap 95% upper bound < 1 (10,000 whole-network resamples), at least 39/64 wins and no failures, against their registered comparators on development and final blocks. The earlier B600 (#334717) final gate used ratio ≤ 0.98 with the same interval and win conditions. Scores average official float32 final-layer MSE times $\max(0.1, F/2^{41})$ over networks. B600's final block was separate from that new residual training and selection; project-wide historical non-exposure of those networks was not certified. [E37]

Step	Pool	Updates (selected)	Development vs parent (wins/64)	Independent final vs parent (wins/64)	Official
#334717	256	600 epochs, auxiliary	0.9574 adjusted (62; vs #334316)	0.9580 (56) on full 256-319	3.3918951472e-9
#334793	256	+28,800 terminal (1,152)	0.9935 (41) vs #334717	0.9909 (42) on full 320-383	3.3726626408e-9
055d	384	+5,760 terminal (2,592)	0.9913 (48) vs #334793	0.9935 [0.9905, 0.9966] (37) on full 544-607	not submitted
#335413	760	+11,400 terminal (9,405)	0.9936 (42) vs 055d	0.9901 [0.9857, 0.9944] (44) vs #334793; 0.9940 [0.9910, 0.9969] (44) vs 055d, on full 448-463 and 480-527	3.3425257575e-9

The official ratios reproduced the local final ratios: #334793/#334717 = 0.9943 against 0.9909 locally (0.34 points), #335413/#334793 = 0.9911 against 0.9901 (0.10 points). The 055d row is the gate working as intended: its development gain (48/64) shrank to a 37/64 win count on the sealed block although the ratio and its interval still favored it, and a second independent block (448-463 and 480-527, scored for the #335413 comparison) gave the same picture for 055d against #334793 (0.9961, 38/64). We kept the wins criterion and did not submit it. [E37, E38]

What the lever is and is not

At this operating point, expanding the training pool and changing the objective helped; the tested capacity changes did not pass their gates. The following studies used their respective registered blocks and gates (all in [E37, E38]): hidden width 128 against 64, ratio 1.0021 (4/64); an additional 8-dimensional signed transported state (593 extra parameters, fixed 2,880 updates), 0.9987 against 055d with 36/64 wins and an interval reaching 1.0005; a second message channel retrained on the whole model, 1.0007 against the 64-wide model; a final-layer-only variant and a GRU variant, both worse at the pilot stage; three equal-weight seeds averaged, 0.00015 better than one (the seeds predict almost identically); the auxiliary all-layer objective continued at matched updates, 0.9958 (43/64) against the terminal objective, below the gate. Data gave -0.87% from 256 to 384 networks and -0.64% (development) / -0.60% (final) from 384 to 760, while the remaining held-apart or excluded public networks were not available for unrestricted training under this protocol. The two tested horizon and learning-rate continuations also failed to improve the selected checkpoint: on the 760-network pool, continuing the unchanged recipe to 240 data epochs (22,800 updates) never produced a development score below the point selected at update 9,405, and continuing from that point at learning rate $1e-4$ for 60 epochs only drifted upward ($3.413e-9$ to $3.424e-9$); neither run produced a candidate, and the last protected block (full 16-31 and 528-543) was left unopened. The fixed last state of the 760-network run (0.9964, 39/64 against 055d) is reported alongside the selected one; selection by scheduled development score was preregistered and is the deployed rule. The tested layer-specific capacity expansion also failed its gate after #335413: giving each of the sixteen layers its own copy of the corrector (105,504 parameters, initialised from the shipped shared weights so that the model equals #335413 at step 0, fresh optimizer, 120 epochs on the 760-network pool) lowered the training loss monotonically while the development score rose monotonically from the third epoch on; the best checkpoint (epoch 2) was 0.9985 against #335413 with 38/64 wins and a paired-bootstrap interval [0.994, 1.003], short of the preregistered 0.995 and 39/64, and was not built [E41].

The 760-network run needed one engineering change only: the 64 MB float32 weight tensor of every training network is regenerated once from the official seed, digest-checked against the collection record, written to disk and re-read on demand, so that neither host nor GPU memory holds the pool (49 GB). The trainer is the frozen pilot code; on the 384-network pool the new loader reproduces the pilot's recorded trajectory bit for bit at update 96 (selection score, MSE and training loss).

Timing and the grader

The corrector adds a fixed sequence of client operations after the frozen estimator returns, so its wall and residual cost is measured, not derived. On a quiet host (no other numerical process; the screen records every process's CPU before and after each leg), six baseline/baseline pairs of the frozen estimator on the same public network gave a wall-time ratio of 0.9997 with 90% interval [0.9974, 1.0023] and a residual difference of -0.6 ms with interval $[-1.5, +0.3]$ ms (the preregistered equivalence bands were [0.97, 1.03] and ± 15 ms); eight baseline/candidate pairs then gave a candidate wall ratio with 95% upper bound 1.0138 and a residual difference with upper bound $+8.0$ ms, i.e. projected maxima of 108.5 s and 0.355 s against the 114 s and 0.38 s planning margins. Earlier attempts at the same measurement under other workloads had failed the baseline/baseline band (wall interval [0.911, 1.026]); the lesson, consistent with Section 5, is that local wall ratios are only interpretable on an otherwise idle host, and that the quiet-host block must be run before, not instead of, the official grade. #334717, #334793 and #335413 have confirmed GRADED status and public aggregate scores. Aggregate raw-MSE agreement cannot rule out failures on sealed networks; their complete official per-network timing and failure records were not harvested in this account. A numerical finding from the same work: the installed metering library executes multi-operand `einsum` through an optimized pairwise path, so a reference built on plain NumPy with the default contraction order differs from the metered result in the fifth significant digit of some outputs; bit-exact references need the same contraction order. [E39]

Transfer to a public engine (16-18 October)

The corrector reads only quantities that a cumulant-propagation estimator computes anyway, so we asked whether it transfers to an engine we did not write. The engine is LordProcciuto's `e30_a` (submission #335981, graded 3.1162994498e-9 with raw MSE 2.0553951785e-8), released under the MIT licence and itself a modification of 504aldo's MIT-licensed V29; both notices are kept at the top of the submitted file and our changes are marked there. `e30_a`'s own per-layer diagnostic capture supplies 15 per-neuron quantities (pre-activation mean and variance, the four Hermite readout terms, the transported third- and fourth-cumulant diagonals, the (2,1)-slice and covariance row sums, the baseline and its mean correction). With 16 derived ones (among them the standard deviation and standardized mean, the pass probability and density, the layer's fourth-cumulant coefficient, normalized cumulants, the returned row, the column energy of the weight and layer flags) the corrector keeps the Section 6 architecture (two transported scalar messages, a $33 \rightarrow 64 \rightarrow 64 \rightarrow 2$ GELU network, 6,466 parameters) and is retrained from scratch on that state. On full 384, the saved capture check reports identical prediction bytes with and without capture; on that same network, a build with the corrector switched off is byte-identical to `e30_a` in prediction and metered FLOPs. Only the final-layer mean changes.

The data split and statistical gates were fixed before training. Deployment gate 3 was amended once before training and again after the pilot but before the full run; the selected checkpoint was committed before the final opening. `e30_a` was run once on 824 public full-split networks (no failures): 696 for training, full 384-447 for development, and full 900-963 as a final block opened once. The training recipe is the Section 6 one (AdamW 5e-4, weight decay 1e-4, clip 1, batch 8, an auxiliary all-layer term decaying to zero by epoch 60, terminal objective, 150 epochs, best development epoch). A 207-network pilot (development 0.982) checked the code path before the full run.

Check	Criterion	Result
Development raw MSE, full 384-447	$\leq 0.980, \geq 40/64$	0.9732 [0.9665, 0.9799], 56/64

Check	Criterion	Result
Final raw MSE, full 900-963 (opened once)	≤ 0.985 , $\geq 38/64$, interval below 1	0.9780 [0.9713, 0.9848], 49/64
Later holdout score, full 544-607	Recorded cost approximation; fresh for the e30_a project	0.9764, 53/64
Resources, paired and interleaved	wall ≤ 1.03 , residual $\leq +0.02$ s	wall median 1.009, residual median +0.0008 s (3 pairs, full 400); FLOPs +0.144% on full 400, +0.153% on full 384
Official #336611	graded	3.0436118103e-9; raw 2.0042401836e-8 (score $\times 0.9767$, raw $\times 0.9751$ against #335981; $\times 0.9106$ against #335413)

Deployment gate 3 failed on all four checked development networks: the amended elementwise bound was two units in the last place (ulp), but the observed maximum was 2,112 ulp. Amendment 1 had replaced a relative-correction bound by an absolute $1e-7$ bound before training; amendment 2 replaced that bound by two ulp after the pilot, before the full run and final opening. Near-zero reference values made the latter stringent: the result record reports values around $1e-8$ with differences around $5e-12$. The maximum absolute difference was about $1.5e-8$, induced MSE was at most $1.24e-11$ of the engine's, and deployed per-network MSE agreed with the reference within $1.2e-7$ relative. These small errors do not turn the failed gate into a pass. The protocol said all gates must pass and a failure meant no submission; #336611 was subsequently submitted with user approval despite this failure, and no post-result threshold amendment was made. The local projection was about $3.05e-9$ (3.03 - $3.07e-9$), consistent with the grader. The recorded local development comparison of the engines was 0.930 (e30_a / #335413); the official ratio calculated from their saved scores is 0.9323. The earlier 0.12% FLOP estimate compared against a diagnostic-capture baseline, which itself adds operations; the table instead uses the uncaptured baselines. [E48]

Three follow-ups were preregistered and stopped at different stages. *Seed ensemble*: three further seeds reached development scores 0.9716-0.9727, but the four learn nearly the same correction; on full 544-607 (fresh for the e30_a project, previously consumed in the earlier project) the four-seed mean scored 1.0008 against the single model (29/64; interval [0.9994, 1.0022]), its 0.22% raw gain undone by about 0.30% more bill. These holdout scores use the recorded approximation: captured per-network bills multiplied by fixed corrector increments, rather than deployed bills for every holdout network.

GRU state transport (the GRU arm of Section 6, 27,586 parameters, hidden state carried by the transposed weights): best development 0.9758 at epoch 14, then overfitting (training loss 0.96 to 0.79 while development rose to 1.14 by epoch 150), and the hidden transport adds about 1.0% to the bill. It was stopped on development evidence; its registered H2 holdout gate was not run.

Data scaling: halving the training pool cost the two-message arm 0.0025 and the GRU arm 0.0144, so the registered rule called for more data for the GRU arm. The synthesis log records 621 completed GPU target jobs and 620 completed feature captures, all with exit code zero; 560 synthetic networks entered the interim training run. Networks used the official generator form with private seeds and GPU Monte-Carlo targets with float64 accumulation. Amendment D1, before any synthetic training, retained the first 16 targets at 2^{27} inputs and reduced subsequent targets to 2^{26} . The validation record covers only two official networks at

2²⁵ inputs: discrepancy from the official 10⁹-sample targets was 0.983-0.999 of predicted sampling noise over all layers. This limited check is consistent with the noise model; it does not establish absence of bias across all networks or sample counts. At 1,256 training networks the GRU arm reached 0.9715 (epoch 13), against an extrapolated 0.9635 and the registered continue threshold of 0.970, and overfit afterwards. Generation was stopped, and the further block full 448-463 and 480-527 was collected but never opened. [E49, E50]

The reading is limited to these runs. The two-message corrector moves to a different engine with a smaller relative gain than its cumulative gain on our own chain, plausibly because e30_a already applies its own per-layer mean-correction table. The extra capacity we tried did not transfer from 696 networks, and the interim data expansion bought about a third of what the learning-curve extrapolation suggested on development. The mean-correction-table explanation was not tested by an ablation. Only the corrector and its training pipeline in #336611 are ours. The organizers require substantial original changes and reserve the final decision on prize and ranking eligibility; attribution alone does not establish eligibility. [E48]

7. Negative results and limits

Independent uncertainty and a later public result

The alternative-basis geometry was first tested with the previous coefficient table on a frozen independent block of 32 full-split networks (J, full672-703): adjusted-score ratio 0.9997 with 18/32 wins and a paired 95% bootstrap interval [0.9987, 1.0006] (20,000 resamples), so that statistical gate failed. Combined later with the current K4 table, the same geometry improved the official score over its direct parent (#331354). Both observations are retained: the later official success does not convert the failed independent block into a success, and no table was refitted against official scores. [E3, E9]

Sibling refits and selection on the public set

On 23 September nine configurations fitted to the same 1,360 native responses (full32-47) were graded: different ridge values, relative weighting, and single-axis versus joint coordinate sets. All nine completed 100/100 within resource limits. Three improved on #331836; #331910 was the best by 0.180%, and #331905 (relative weighting) was within 0.005% of it. Every single-axis variant (K3 only, K4 only, K22 only) was worse than the previous best. Because the winner was chosen among nine grades on the same 50 public networks, its advantage is subject to selection. A paired normal approximation over those 50 networks gives an illustrative 95% interval of roughly -0.47% to +0.12% relative to #331836. No coefficient, weight or ridge was re-tuned against official scores.

Other measured failure modes

- **Rank cap in both directions.** Reducing the carrier cap from 7,680 to 7,168 saves metered FLOPs (multiplier -1.2%), but after a full matched refit on the new trajectory (1,360 responses, three frozen ridges) the best candidate was still 6.07% worse on development networks (1/8 wins, worst +12.4%); refitting recovered only about 0.19%. Raising the cap on the deployed arithmetic (26 September, shipped tables) was a wash at 8,192 (raw 0.961×, bill 1.044×, adjusted 1.001, 4/8 wins) and a loss at 8,704 (raw 0.963×, bill 1.088×, adjusted 1.048): the raw gain saturates while the bill keeps rising, so 7,680 beat the tested neighboring caps under those recipes. The two tested caps near the 0.1 cost floor were also worse: at cap 4,096 the bill reaches 2.03e11 (multiplier exactly 0.1, truncation from layer 5) but raw MSE after a full matched refit and a six-coordinate response fit is 9.38e-8, 3.6× the deployed value, so the floor score would be 9.4e-9; cap 4,608 (F/B approximately 0.0999, clipped to a score multiplier of 0.1) gives 7.05e-8. Early truncation is the expensive kind. [E22]

- Learned students at the cost floor.** Two distilled estimators were trained on a public 2,048-network teacher corpus (final means at 1e8 samples plus per-layer teacher states) to test whether a floor-cost learned model could compete: a 32-state-per-neuron recurrent student (1.0e10 FLOPs) collapsed once teacher forcing was withdrawn (free-run holdout MSE 3.6e-2, identical weights from epoch 7), reproducing the autoregressive instability seen in Phase 1; a K2 Gaussian covariance chain with a learned readout corrector (8.1e10 FLOPs, no learned quantity fed back into propagation) reached raw MSE 1.26e-6 on mini 80-87 and 1.32e-6 on full8-15, 25× above the 5e-8 smoke gate. Neither was submitted. [E23] Relaxing the retained-coordinate schedule by 64 coordinates on layers 9-14 raised the bill by 5.3% and raw MSE by 9.2% (0/8 wins): the tables are fitted to the pruned trajectory, and loosening a design choice they were matched to worsens the readout. [E20]
- Low-rank fourth-order core.** Eigen-truncating the exact AUGMENT-kind fourth-cumulant core of the reference implementation [1] diverged at ranks 16-64 and was 8× worse even at rank 192. This was measured at width 256, depth 16; its application to the width-1024 contest shape is an extrapolation. Those tested eigen-truncations did not preserve the reference core's accuracy; the result does not exclude different ranks, fitted cores or other structures. The AUGMENT-kind chain itself (matrix fourth-order core with the (2,1,1) and (3,1) diagrams) is rank-expensive in a factorized carrier: at width 256, depth 16 its uncapped third-order rank is 33,024 columns against 7,680 for the simple kind, and under a fixed column cap equal to our operating fraction (0.44 of the simple rank) its accuracy advantage collapses from 0.34× to 0.92× (norm truncation), 0.89× (readout-sensitivity truncation) or 0.93× (age-tiered shared subspace), while protecting the AUGMENT-born columns is infeasible because the (2,1,1) feed alone exceeds the cap by layer 6 (3e8-sample references, 95% intervals ± 0.014). None of these four tested truncation policies preserved the uncapped advantage at this width and column cap; this does not identify the leaders' representations or rule out other column-capped methods. [E24]
- Oracle ceiling of pair fourth-order blocks.** Injecting 1e9-sample reference values of κ_{4_ijij} at every layer on 8 fresh networks improved final MSE by only 6.4% (8/8), with shipped tables and an exact dense carrier. This measures the effect of substituting those reference blocks on this trajectory; it is not an upper bound on all fitted or jointly modified block models.
- Wall-time kernels.** Five alternative kernel arrangements produced wall-time ratios of 1.00×; in our own configurations the faster wall times came only from shallower multiplication recursion, which costs FLOPs.
- Hybrids with a public entrant's estimator.** The MIT-licensed Phase-2 estimator released by entrant 504aldo (repository [504aldo/whest-p2-cumulant-k3](#), version V29, commit [18c17e2d](#)), which regenerates a matrix fourth-cumulant core each layer and carries older third-order sources in a shared subspace, has 0.823× our raw MSE on the eight mini development networks at 1.82× our bill (multiplier 0.254), i.e. a worse adjusted score. Transplanting our gate-dead-neuron pruning into it reached multiplier 0.241 at raw 2.33e-8 (gate: 0.215 and 2.30e-8); a fixed 50:50 average of the two estimators has error correlation +0.65 where about -0.2 would be needed to pay for two chains (adjusted 2.06×, 0/8); an output-space basis for its old tier improved on its reduced-rank variant (0.974, 7/8) but was 1.37× worse than our incumbent; reduced rank plus an eight-coordinate active correction failed its linear gate (0.989 vs 0.98). Finally, that estimator's documented "memoryless fourth-cumulant regeneration" (a per-layer scalar law $G = \text{diag}(g) + \lambda \cdot C$ for the fourth-order matrix core, reported by its author as roughly halving raw MSE at no cost on two networks) was re-run in the author's own lean chain on full8-15: ON/OFF ratio 0.926 (7/8 wins, worst +9.0%), with an OFF mean of 3.12e-8 already above our deployed raw MSE. The 26 September symbolic bound put a shared-leg formulation inside our carrier at at most +5.5% of the bill, while the donor re-run failed to reproduce the claimed accuracy gain. [E21] A projected shared-feed implementation was subsequently built and tested with matched refits; its completed negative result is reported separately below. [E25]

- **Stronger pruning at the middle plateau.** Delaying the retained-coordinate schedule until the existing 832-wide plateau on layers 9-12 cut the bill by 3.8% but raised raw MSE by 18.0% (0/4 wins). A 768-core-plus-64-fringe decomposition of the 832-wide transports was rejected on paper: even with all fringe linear and copy costs deleted, the eight products cost 0.64% more than the incumbent.
- A 56-parameter head improved its surrogate training objective, but actual response disagreed with the surrogate by 15.436%, above the 5% implementation-transfer gate. This was not evidence of a held-out improvement. [E5]
- A nonterminal output-row policy reduced cost by 6.63% but increased the eight-case adjusted-score ratio to 1.72599 (0/8 wins). It affected diagonal readouts, cap normalization and downstream state together, so it was not a single-mechanism ablation. [E6]
- Official experimental #330935 has a saved FAILED status; the retained page observation did not expose the complete traceback or certify the detailed cause. A related local implementation exposed tuple/transpose and handle problems, but the full official traceback was not available, so the remote root cause remains unconfirmed.

Bill cuts that failed the residual gate (2 October)

Three bill cuts held on 1 October had failed a conservative local forecast (alternating client/server sessions paired against SFP, scaled to the official #333203 residual): forecast maxima 0.4237 s (ITN), 0.4239 s (HFP) and 0.4089 s (HBR), against the 0.4 s gate. ITN, submitted once to test the forecast, graded with no failures at an official maximum of 0.387 s (#333546). We read this as showing the rule to be about 36 ms pessimistic and submitted HFP and HBR. Both failed with RESIDUAL_WALL_TIME_EXHAUSTED on sealed networks: #333552 (HFP) on 4 at 0.404-0.440 s, #333553 (HBR) on 3 at 0.400-0.406 s. Those two were graded concurrently; HFP plus ITN graded alone (#333561) failed on 8 at 0.401-0.414 s, showing that concurrent grading was not necessary for failure. The "pessimistic forecast" reading is withdrawn: for HFP and HBR the strict forecast was right, and one passing ITN run does not calibrate it. With all failures sealed, the three public scores (3.5848-3.5851e-9) look like improvements, but a failed network in private evaluation scores as a zero prediction with no discount, so none can be nominated. [E31]

Projected shared-feed regeneration with matched refits (27-29 September)

The shared-leg path was subsequently built on C47 at width 1024, depth 16 and cap 7680. A is unchanged C47; B adds a matched refit; C combines regeneration with its own refit. Both fits used full32-63, 28 K3/K4 coordinates, ridge 0.5 and one bounded update, retained on strict native training-loss improvement. Their fitted coefficients differ. Four synthetic design networks checked MC precision before one evaluation on eight separately seeded synthetic development networks. [E25]

The frozen development gate required both C/A and C/B score ratios at most 0.99, at least 6/8 wins and worst-network ratio at most 1.10. The result was:

Development comparison	Adjusted-score ratio	Raw-MSE ratio	Wins
B / A	1.00375	1.00375	1/8
C / A	1.23587	1.21073	0/8
C / B	1.23125	1.20621	0/8

Ratios divide sums across networks; raw MSE removes each saved score's own cost multiplier. C added 2.055-2.078% FLOPs. Within-MC relative half-widths were 0.1688% and 0.1679%, below the 0.5% limit. Precision, cost and integrity passed; accuracy failed. The 16 confirmation networks remained unopened and no official submission followed. The finite synthetic sample does not guarantee leaderboard coverage.

With tables identical to A, C's training MSE was already 23.7229% worse; after refitting, C/B remained 23.2629% worse. C's update recovered only 0.5072% of its own MSE. Thus new coefficients or development-set transfer alone cannot explain the initial loss. Saved C/B predictions first differ at output layer 2; RMS displacement grows from $9.13e-6$ to $7.73e-5$ at layer 16. This is not layerwise error: intermediate truth was not used. Diagonal K4 and refit changes can affect layer 2 before transported nonzero feed affects layer 3.

For final predictions, let $d = C - B$. The identity $MSE_C - MSE_B = \text{mean}(d^2) + 2 \text{mean}((B - \text{truth}) d)$ gives saved-result terms of **+20.989% - 0.368% = +20.621%**, relative to B's estimated MSE. The interaction is inferred from saved MC-based MSE differences, without a new confidence claim. C changes diagonal fourth-cumulant projection, power-cumulant feedback, shared feed and newborn updates together. The result rejects this combined recipe under the registered fit and operating point; it does not isolate a component's causal effect. [E25]

Two preregistered gates (2 October)

Two remaining representation changes suggested by public material were each given a short gate with thresholds, controls and falsifiers frozen before any run [E27]. Both were killed.

Gate A: lifetime truncation of AUGMENT feed-born columns. In the reference AUGMENT chain [1] the (2,1,1) feed creates new third-order columns at every layer, which is why it is rank-expensive under a cap (E24). The hypothesis was that their value is concentrated in their first one or two transports, so that discarding feed-born columns after k transports, with ordinary columns kept under the norm cap, would retain much of AUGMENT's accuracy without its rank growth. The operating point was that of E24 (reference factorized chain, width 256, depth 16, seed 0, cap 3,604 for both kinds, the E24 3e8-sample references with split-half noise subtraction). GO required $\text{ratio_1} = \text{MSE}(\text{AUGMENT}, k = 1) / \text{MSE}(\text{SIMPLE}) \leq 0.70$ with jackknife upper bound ≤ 0.75 and a pessimistic cost-adjusted ratio ≤ 0.85 ; KILL was $\text{ratio_1} > 0.80$ or a lower bound above 0.75. Six control runs reproduced the 27 September prediction hashes, and all three E24 anchors were reproduced. The ratios were 1.610 ($k = 0$), **1.236** [1.218, 1.253] ($k = 1$), 1.126 ($k = 2$), 1.006 ($k = 3$) and 0.919 without a lifetime limit (the norm-capped E24 policy): KILL. Shorter life is uniformly worse, the opposite of the hypothesis; even dropping only the roughly 100 columns per layer older than three transports costs 9.5% against the uncut policy. Our post-hoc reading, not part of the gate, is that feed-born columns act as cancellation partners of the same-layer (2,1,1) contributions to the fourth-order core, so truncating them breaks a pairing rather than removing a small remainder. Truncation also buys almost nothing: total FLOPs at this cap were 4.07-4.10× SIMPLE for every k (1.17× counting retained third-order columns only), because the (2,1,1) machinery does not shrink with column lifetime. [E27]

Gate B: a public fourth-order scalar law with lambda fitted to the final mean. The public entrant's estimator discussed above (E21) replaces the off-diagonal of the matrix fourth-order core by the per-layer law $G_pre_off = \lambda I C_pre_off$; through a (3,1) Wick term (internally WK431) the regenerated core also enters the pair blocks P11, P21 and P22. In E21 lambda was fitted to an internal fourth-order target in the author's chain, and in E25 the author's table (v17) was used unchanged; lambda had never been fitted to the final mean. Gate B fitted lambda to final-layer MSE on a float64 replica matched to the deployed chain (cap 7,680, retained-coordinate schedule, shipped tables, pair terms and dressing), training on mini networks 40-59, with the shipped covariance-shaped template (A, Section 2) as comparison. Frozen steps: B1, a global

scale of v17 from -0.5 to 1.5 , passed if the best scale improved on scale 0 by at least 3%; B1b, per-layer lambda from a finite-difference Jacobian with ridge chosen by leave-one-network-out, passed only if the actual leave-one-out MSE was $\leq 0.97 \times A$ with at least 14/20 wins; B2 (matched refits, GO at ≤ 0.95 with $\geq 12/16$ held-out wins and worst ≤ 1.03) was conditional on B1b. [E28]

B1 passed mechanically (best scale $0.5 \times v17$, -6.5% against scale 0, 19/20 wins), but no scale beat A: the best was $1.0508 \times A$ (2/20). Per-layer lambda (all positive, mean $0.47 \times v17$) gave an actual leave-one-out MSE of **$1.0408 \times A$ with 2/20 wins**, so Gate B was killed and B2 not run. The first-order model predicted the actual forwards within 0.2%, and per-layer freedom added only about 1% over the global scale (0.9905, 17/20), a small measured gain under the tested parameterization rather than a collapse of the coefficients. Moving lambda only into the core transport without the (3,1) term gave $1.0099 \times A$; adding that term was harmful at every profile tested. Single-path exclusions on held-out networks 80-87 (tables and pair terms off, v17 scale, relative to the template) located the harm: the full term gives 1.307; excluding its share from the next layer's covariance (k11) gives **0.9914**, removing 103% of the excess; excluding it from the fourth-order channel (1.327) or from the carried third-order memory (1.324) does not help. With our pair terms on, the same term costs 1.136 rather than 1.307, so those terms partly mask the damage rather than replace the term. This does not test the author's chain, where the law is reported to help; it shows that in ours, with tables fitted to our trajectory, the term's contribution to the next layer's covariance is harmful at every scale tested. [E28]

Six further gates and two rechecks (3-4 October)

After the call-count work freed residual time, we asked whether any representation or bill lever had been blocked only by that gate. Each gate below froze its thresholds and controls in the repository before its first run, used the reference factorized chain [1] at width 256 and depth 16 with the E24 operating point and split-half-corrected Monte Carlo references, and reproduced the 27 September prediction hashes for its fixed-cap controls. The score proxy is corrected final-layer MSE times cost, relative to SIMPLE at a fixed cap. [E33]

Gate C: a machinery-free cost bound for AUGMENT. We charged AUGMENT only for its third-order columns and assumed every AUGMENT-only operation (the $(1,1,1) \rightarrow (2,1,1)$ and $(2,1,1) \rightarrow (2,1,1)$ maps, the matrix fourth-order core, the extra nonlinear terms) cost nothing, sweeping nine AUGMENT caps from 2,048 to uncapped against five SIMPLE caps from 1,536 to uncapped. GO required a score proxy ≤ 0.85 of SIMPLE at cap 3,604; the best AUGMENT point was **0.951** (cap 5,120), or 1.39 against SIMPLE's own optimum at cap 2,048: KILL. Because arithmetic can only lower the machinery cost toward the zero already assumed, eliminating those machinery costs alone would not meet the registered 0.85 gate for the tested caps and measured errors. Between caps 4,096 and 8,192 AUGMENT's error falls in proportion to its column count (proxy 0.95-1.02), so its advantage is bought column for column. The source shows why: every term AUGMENT adds to the third cumulant has the path form $A_{ij} \cdot B_{jk}$, which after one transport becomes a general rank- n factor, which explains the cost of this factorized implementation but does not exclude a different representation or a cheaper construction. [E33]

Gate D: a plane-only third cumulant. Before every transport we replaced the factored third cumulant by its repeated-index part (diagonal and (2,1) plane), re-factored at rank n with the reference's own routine, so the third-order cost no longer grows with the cap. GO required a proxy ≤ 0.85 against SIMPLE at cap 2,048. SIMPLE with the plane only had **5.40×** the error at $0.45 \times$ the reference FLOPs (proxy 2.46); AUGMENT with the plane had $4.07 \times$ the error at $3.75 \times$ the FLOPs (proxy 15.3): KILL for both. The all-distinct entries carry the information that later layers need, consistent with a September analysis showing that the plane alone cannot be an exact closed state, and with the earlier finding that dropping whole coordinates is tolerated while structural (subspace) truncation is not. [E33]

Gate E: an adaptive rank. Leading entries' metered FLOPs vary across networks and anti-correlate with ours (next subsection), consistent with a data-dependent rank. We compared the fixed cap 2,048 with a threshold rule keeping the smallest top set holding $1 - \text{eps}$ of the summed column scores ($\text{eps} = 0.047$, frozen on seed 0 to match the fixed cap's cost), on seeds 1-7 with new $1.02e8$ -sample references. GO required a geometric-mean proxy ≤ 0.97 with at least 5/7 wins. The result was **1.0023 with 3/7 wins**: about 3% fewer columns and 3.5% more error, scattered 0.90-1.10 across networks: KILL. Together with the per-network cap oracle ceiling of 1.46% measured in September, these tests do not explain the leaders' advantage; they do not identify the leaders' algorithms or exclude other adaptive rules. [E33]

Gate F: AUGMENT's path legs without the matrix core. Part of what AUGMENT adds to the third cumulant needs no fourth-order core: two-leg paths through the third cumulant's (2,1) slice, a doubled covariance edge, and the fourth cumulant's (2,2) slice, all built from quantities SIMPLE already carries. We added these legs to SIMPLE (the leg list of a copy of the reference routine switched by flags; with no flags it reproduces the 27 September hashes) at caps 2,048, 3,604 and 5,120, with the score proxy of Gate D and thresholds $\text{GO} \leq 0.85$, $\text{KILL} > 0.95$. The (2,1) leg alone was best: at cap 2,048 it lowered error by 8.5% for 3.7% more columns (proxy **0.945**); the doubled edge and the (2,2) leg did not help, alone or combined. The verdict is WEAK (between the 0.85 GO and 0.95 KILL thresholds). An exploratory check on seeds 1-7, not part of the verdict, gave error ratios from 0.83 to 1.14 (geometric mean 0.958, proxy 0.989, 4/7 wins): a small average gain that is not consistent across networks. Two follow-ups decided the port. Gate F2 re-expressed the legs exactly as symmetric columns of the deployed column form (for every middle index r , the newborn form becomes $Q_r = w_2 a a^T + a g^T + g a^T$, rank 2, written as two columns $\lambda F(x) 3e_r(x) F$); its controls reproduced SIMPLE and the general-CP legs to $4e-14$ and $7e-16$, and on seeds 0-7 at cap 2,048 it gave a geometric-mean error ratio of **0.877** (proxy 0.898, 6/8 wins), better than the general-CP form, so its preregistered verdict was to port. [E34]

On the float64 replica of the deployed chain (held-out networks 80-87, deployed caps, pruning, pair terms and template), the legs kept that direction only while the newborn leg vector F_1 was free: with the same per-layer column counts, the error ratio was **0.925** with tables off (8/8) and 0.951 with the shipped tables (7/8). But the deployed carrier never transports F_1 ; it stores $F_1 = \alpha P + \beta X$, so a free F_1 costs one more $n \times n \times R$ transport per column, about +38% of the bill. Keeping the deployed projection leaves about 1% (0.994 and 0.990, 5/8). Representing the legs exactly in that column form needs three columns per middle index (a remainder vector and two leg directions cannot share two columns), and with those extra columns the error was $5.9\times$, $2.4\times$ and $1.2\times$ the deployed error at caps 7,680, 10,240 and 12,800, for proxies of 6.9, 3.6 and 2.1. The port was stopped. The finding is that the deployed carrier's economy comes from tying F_1 to the two transported vectors, and that tie is exactly what excludes the one diagram family that improved accuracy in the reference. A last variant kept F_1 free only for the first k transports of each newborn column and then projected it: the error gain grew steadily with k (0.995, 0.989, 0.951 and 0.919 with the shipped tables for $k = 0-3$, 8/8 wins from $k = 2$), so the legs' information is used over several layers, but the extra transport grew about 4.5% of the bill per layer of lifetime and $k = 1-3$ had cost-adjusted proxies of 1.036-1.041; $k = 0$ was best at 0.995 but missed the 0.97 gate. The reverse assignment, which costs nothing, keeps F_1 exact and projects the remainder vector X onto the span of the transported column and F_1 ; it raised the error 19.7-fold with the shipped tables and 14.7-fold without them (0/8 wins in both), so X carries far more of the answer than the leg direction. This confirms the deployed choice of projecting F_1 rather than X , and with the free- F_1 , three-column and lifetime results none of these tested variants retained both contributions at a qualifying cost in two transported vectors per column. [E34]

Gate G: the (2,1,1) feed of our own fourth-order core into the third cumulant. SIMPLE omits the diagram in which a fourth cumulant with one repeated index creates third-cumulant mass at the nonlinearity. With the deployed covariance-shaped core it is a vector-times-matrix source $y(x) M$ at each layer, $M = D C_{\text{off}} D$ (D the

first Wick coefficients, C_{off} the pre-activation covariance off the diagonal); with the isotropic core the matrix is the metric $W W^T$ instead. Cost is not the obstacle: written on the existing transported legs the covariance-shaped feed adds about 0% of the bill and the metric-shaped one about 5.7% (for $W W^T$), against +14-23% if the matrix were carried and transported separately. On the float64 replica (held-out networks 80-87, deployed caps and pruning, shipped tables, the feed carried as dense sources, avoiding a feed-compression approximation in this test) the physical sign made the error $1.228\times$ (0/8) with the covariance shape and $1.041\times$ (0/8) with the metric shape. Only the sign-reversed covariance arm helped, $0.977\times$ (7/8), short of the preregistered 0.97 and an unphysical correction rather than the diagram; a matched refit on an earlier chain (14 September) had given the same picture (+3% to +22% with the physical sign, -2.1% to -2.7% reversed). The diagram that the public estimator's matrix core feeds through therefore does not carry over to our core, in either shape; this is consistent with the regenerated-core results above (E21, E25, Gate B). [E35] A later check on the un-fitted chain (learned tables and pair terms off, with and without the rank cap; the same 8 networks, 12 October) gave the same picture, $1.145\times$ with the covariance shape and $1.026\times$ with the metric shape, 0/8 in every arm: the feed used alone is harmful independently of the fitted tables and of truncation, consistent with the reference ablation in which removing either the (3,1) or the (2,1,1) diagram from the full AUGMENT set diverges. The AUGMENT gain belongs to the complete diagram set, which was never built on this carrier because of its cost (Gate C). [E40]

Gate H: a covariance correction fitted to the final mean on forty networks. The pre-activation covariance is the most accurate object our chain carries, yet errors in it are amplified on the way to the final mean. A correction to it fitted to the final mean (not to the covariance's own truth) was tried once before (1 September) with 128 coefficients on four training networks and overfitted: 0.885 on the training networks, 1.055 on held-out ones with the same coefficients, 0.988 (5/8) after strong ridge and a matched refit. Here the off-diagonal covariance at layers 1-14 receives $\Delta C = \sum_k \theta_{\{g,k\}} B_k$ over seven symmetric, zero-diagonal matrices that cost $O(n^2)$ each: O and its elementwise square and cube (O the off-diagonal covariance), the symmetrized (2,1) slice, $md\ md^T$, $\mu\mu^T$ and the all-ones matrix, each off the diagonal. Each is scaled to the Frobenius norm of O and damped by $g = q/(q+10^{-4})$, q the relative size of the raw diagonal third cumulant, so the correction vanishes for Gaussian or uncorrelated states. Coefficients are shared within four layer groups (28 in total) under a trust bound $\sum_k |\theta_{\{g,k\}}| \leq 0.02$ per group. The corrected covariance rides on the existing $W C W^T$ product, so no new n^3 product is needed (planned bill about +0.39%). We fitted θ on mini networks 40-79 on the float64 replica (deployed caps, pruning and shipped tables) with a central-difference Jacobian, five network-level folds and λ by the one-standard-error rule. The first attempt stopped at the preregistered step-size check: two step sizes disagreed by 12% on one coordinate, because perturbing the covariance changed which columns the rank cap kept at every capped layer. A preregistered amendment replayed the baseline's cap selections and keep masks inside the Jacobian only; all checks then agreed within 0.05%, and every evaluation kept the real selections. A ten-network pilot gave an out-of-fold ratio of 0.963 (10/10). The forty-network fit gave 0.954 out of fold (38/40), a linear prediction of 0.951 against 0.951 measured, and on held-out networks 80-87, with the shipped tables unchanged, **0.966 in raw MSE (8/8; 0.953-0.980 per network)**. The preregistered pre-refit screen asked for a score proxy ≤ 0.95 with the bill charged at a conservative +1%; the measured 0.976 failed it, so the matched refits and the fresh confirmation networks were not run. (The screen was set before the effect size was known and was not aligned with the later amended matched gate of 0.97.) This is the first change since the arithmetic campaign that lowered replica error consistently on held-out networks, so we measured it on the native estimator under preregistered amendments (frozen thresholds for each step, results recorded in E36). Ported into a copy of the deployed source with the shipped tables unchanged, it kept a smaller but uniform gain on the same development networks: score 0.987 (8/8), with the replica gain shrinking by 1.1-2.7 points on every network. We take this as overlap with native-only steps that the replica lacks, without having isolated it. The first port exceeded the grader's 800 live-array limit (the deployed estimator already peaks at 794). A buffer-borrowing

rewrite then needed a second fix to reproduce the first port's float32 accumulation order exactly; the final port is bit-identical to it on all sixteen layers, peaks at 794 arrays and adds 1,358 client calls. On eight fresh networks of the official full split (512-519, never used before), it gave **0.983 in score (-1.7%), 7/8 wins, one-sided t upper bound 0.994**. Submitted as #334003, it failed six of the hundred graded networks (three wall-time, three residual-time) in a grading run where three of five workers died and grading took 42.5 instead of 35 minutes. The networks that completed show the cost: about 1.7% more wall time and 6 ms more residual time than #333716, so the slowest networks crossed 120 s and 0.4 s. Moving Strassen level-6 work back to level 5 to regain that margin cost the gain itself: wall-time parity left about -0.6%, inside our local timing noise. Every such level change also added arena views and exceeded the 800-array limit. We then rewrote the correction itself to be cheaper, with the same mathematics (closed-form norms for the rank-one bases, one-pass sums, a Horner combination; equivalence checked line by line and against a measured rounding floor). The first rewrite cut its full-matrix passes from 36 to 16 per layer, but its client calls rose slightly (+1,386 against +1,358 for the first port), and its wall time stayed 0.8% above #333716 (local median, with an upper bound of 2.1%). A second rewrite cut the calls to +896 and brought residual time within 7 ms of #333716, yet wall time stayed 1.1% above. A single-core profile of #333716 accounts for the remainder directly: local wall time is 33.0 s of backend work plus 4.0 s of per-operation overhead (about 0.17 ms per client operation), and the second rewrite still adds sixteen full-matrix passes on each of fourteen layers (about 0.7%) and 658 operations (about 0.3%). The remaining cost is therefore the correction's own work, not a downstream effect. We then kept the correction only on layers 8-14 (development score 0.987, 8/8) and removed 770 operations from the deployed Strassen recombination by replacing staging copies with views and batching the terminal copies; predictions stay bit-identical. In 21 interleaved single-core rounds against #333716 (A/A control 1.003, 90% interval 1.000-1.006), the copy reduction alone left wall time unchanged (1.001, 0.999-1.006) while cutting residual time by 12 ms; with the layers 8-14 correction on top, wall time was 1.008 (1.006-1.012) and residual time 7 ms below #333716. The preregistered local wall gate (≤ 1.000) was not met. Submitted as a grader probe (#334316), this version graded **3.541×10^{-9} , -1.2% against #333716 (raw MSE 2.578×10^{-8}), with no failed networks**: the slowest network took 106.6 s (#333716: 106.5 s) and the largest residual time was 0.347 s (#333716: 0.354 s), but per network its wall time was 1.8% above #333716 at the median, against 0.8% locally, so the grader charges the correction's arithmetic about twice as heavily as our single-core runs. Two lessons carry over: on this estimator the residual clock is paid per client call while the wall clock follows the arithmetic, so removing calls cannot buy back an arithmetic cost; and a score-equivalence gate must be set from the rounding noise (here a standard deviation of about 0.0015 per network), not chosen in advance. The correction therefore improves accuracy on unseen networks, and restricted to layers 8-14 it turned into a 1.2% better graded score at the price of about 2% of the wall-time margin. [E36]

Rechecks. Applying the full rank-48 dispatch set that had failed the residual forecast in September (C22) to the #333716 source changed the bill by +0.019% and added 561 operations, because later changes in the line had already taken its saving. Porting our fast kernels onto the public entrant's more accurate chain (E21) cannot close its cost gap: its own ledger puts it at 260 units of $2n^3$ against our roughly 141, and it already uses Strassen-Winograd, while our kernels save 5-10% over Strassen and that chain would need a 34% cut to score below $3.585e-9$. [E33]

Age-gated shared-basis confinement of old third-cumulant blocks (12 October). The deployed carrier compresses the accumulated third cumulant by a global norm cap on its CP columns. A public competitor log (504aldo, repository [whest-p2-cumulant-k3](#), findings F72/F73/F76) compresses the same object by confining the leg vectors of blocks older than four transports to one shared rank- $3n/8$ basis (rank $7n/32$ after seven), and reports +4.6% raw for $\times 0.77$ cost on its chain. We implemented the confinement in the replica (default-off knob, bit-identical when off) and ran a preregistered gate with matched refits on nets 40-79 and

held-out nets 80–87: the deployed cap costs only $1.064\times$ against the untruncated chain once the tables are refitted (the $1.23\times$ of Section 7's earlier analysis was the 31 August chain), so confinement in place of the cap collects at most 6% while costing 7.6% itself ($1.011\times$ vs the deployed chain, 4/8); confinement on top of the cap is a cost lever (third-cumulant work $\times 0.68$) at +6.5% raw (0/8 wins; +7.1% with tensor-energy Gram weights), whose projected score ratio lies between 0.77 and 0.95 depending on how the factored tier products would be billed. All three arms missed their preregistered bounds (damage ≤ 1.06 ; projected score ≤ 0.90 under the ledger and ≤ 0.95 under a dense-billing variant) by 0.1–1.5 points and were not ported. Two independent checks ran alongside: a numerical re-derivation of the Wick coefficients, slices, transport and newborn identity (all at machine precision; the only finding is the metering library's `norm.cdf` losing accuracy below $a = -5$, immaterial to the score) and a code/cost audit whose corrections were registered as an amendment before any result was read. [E42]

The public frontier carrier on our own held-out networks (12 October). 504aldo's MIT-licensed lean chain (`whest-p2-cumulant-k3`), which carries the third cumulant as dense per-source hub legs and regenerates the fourth-order core from the covariance (one scalar per layer), was run on our held-out networks 80–87 against the ARC reference snapshot it was validated with (the May snapshot; against the August snapshot its published numbers do not reproduce — dense AUGMENT 6.9×10^{-8} instead of 2.08×10^{-8} — because the imported coefficient tables changed). Its published values reproduce to all printed digits, and on our networks, with no learned table anywhere: its SIMPLE-class chain is $1.395\times$ our matched-refit deployed configuration (0/8), its regeneration chain $0.816\times$ (8/8), and a dense AUGMENT $0.660\times$ (8/8). In this paired comparison, adding the regeneration channel accounts for the donor chain moving ahead of our control; its base chain is 40% worse than ours. Running the same channel, with the same per-network coefficients, on our chain in pure form (no cap, no pruning, no tables) gives an increment of the same direction (cosine 0.93 at depth) that helps exactly as much as theirs up to layer 5 and nothing at the readout ($3.76\rightarrow 3.76\times 10^{-8}$ against their $3.55\rightarrow 1.97\times 10^{-8}$): the optimal gain along our increment falls to 0.50 by layer 15 and its alignment with our error to 0.37 (theirs 0.96 and 0.67), while against the Monte Carlo truth our third-cumulant diagonal drifts from 2.3% at layer 2 to 6.7% at layer 15 (theirs 2.5% to 4.6%) and our pre-activation variance error is 40% larger. The difference is what the carrier holds: theirs keeps the (2,1)-feedback hub terms as full tensors, ours only their slices (and the deployed carrier projects the leg and caps the columns, which is where the earlier path-leg gate died). The inspected implementations that retain those legs cost about 0.20–0.25 of the budget; no cheaper equivalent construction was demonstrated by this comparison. A 2×2 ablation of the channel on both chains (the $\kappa_4\rightarrow\kappa_3$ birth feed on/off $\times$ the (3,1) use-side term on/off, network 80, the same coefficients) then located the loss. On their chain the two components help additively ($1.965 / 2.133 / 2.145 / 2.289\times 10^{-8}$ for feed+use / use / feed / core). On ours the regenerated core helps ($0.83\times$) and the feed helps further — feed on, use off gives 2.919×10^{-8} , $0.776\times$ our pure base, the first positive transfer of the channel to our carrier — while the (3,1) use-side term costs +29% and cancels both; this is the locus of the earlier Gate B harm, which had never been run without it. The surviving cell (regenerated core plus feed, no use-side term) was then put through the preregistered matched refit on the deployed-matching chain (cap, pruning, pair terms, dressing, tables refitted; two λ tables): 3.0019×10^{-8} and 3.0206×10^{-8} against the deployed control 2.6275×10^{-8} , i.e. $1.14\text{--}1.15\times$ with 0/8 wins (bootstrap 95% CI [1.117, 1.165]), killed by the registered bound of 0.95. What the channel adds to a bare chain, the learned tables and the covariance-shaped fourth-order template of the deployed chain already supply more cheaply; replacing that template with the regenerated core starts 17% behind at identity tables and the feed's -7% does not repay it. Together with the earlier feed-only and feed-plus-use measurements, this rejects the tested regenerated-core combinations under their specified carrier and refit recipes. A final identity-table ablation located the ingredient that reverses the sign: with the cap, the pruning and the pair terms all removed the feed chain still trails the template ($1.074\times$, 0/8) until the F1-leg projection is also removed, and only then leads ($0.954\times$, 7/8); the projection costs the template chain 2.5% and the feed chain 15.4%,

because the feed's content is carried by the exact F1 leg that the deployed carrier projects away, and the rank cap removes the remainder (exact leg at cap 7680: $1.086\times$, 0/8). Among these identity-table ablations, an advantage appeared only on the uncapped, un-projected chain, which costs $\geq 1.37\times$ our compute for a 4.6% gain in raw error. The static λ tables used in the deployed arms are not the cause (pure chain, network 80: 2.92×10^{-8} with the per-network λ against 3.00×10^{-8} with the tables). [E43]

Later checks on the frozen operating point (12-14 October)

Further work did not produce a qualifying replacement for #335413. Uniform parameter averages over the last 20, 40 or 60 epochs failed on the already exposed DEV64 (best ratio 1.0001983 against the selected checkpoint). A pooled hidden-context head failed its conditional TRAIN gate (best ratio 0.9998520); these folds were not independent of backbone training. The external-data pilot completed 32 networks, but its registered projection for 512 networks was 20.5172 hours against a 16-hour budget, so the 512-network refit was not run. This is a cost failure, not a measurement of that refit's accuracy. [E44]

The 13 October study separately tested bidirectional correction, a 7,168-column path with shape-specific arithmetic, and fixed optimizer/regularization variants. The bidirectional suffix model failed its conditional TRAIN64 gate (cost-adjusted ratio 1.0032549 against the frozen model). The initial compressed path saved 0.6537% of prediction FLOPs, below its 1% admission gate; a later geometry-specific kernel saved 1.8271% but its fixed retraining contrast still failed. External-32 reuse, batch accumulation, sharpness-aware minimization (SAM), an initial-weight penalty, L-BFGS, rank-4 adaptation, dropout and a static gradient control variate each failed its registered exposed-DEV gate. Diagnostics of projected messages, target moments, gradient noise and cubic corrections were not official-score evidence. These outcomes apply to the fixed recipes tested, not to all uses of the mechanisms. [E45]

A separate no-training screen used the exact first-layer Gaussian mean as a control variate, transported by the gates of a deterministic mean path (coefficient 1, 8,192 samples, eight already exposed TRAIN networks, four fixed repeats). Its cost was 8.9914% lower than the frozen estimator's, but cost-adjusted error was 3,127.645 times larger (0/8 wins); it also amplified variance against plain MC on the same samples. This rejects that hard-gate control variate, not control variates in general. A mathematical review established conditional small representations and counterexamples, without demonstrating a cheap construction or measured gain on these networks. [E46]

An unsubmitted arithmetic candidate, v4, corrected a client-dtype condition and preserved final float32 bytes in one full native check. The local bill fell from 302,319,812,893 to 300,134,708,125 FLOPs (0.722779%), but local residual time was 0.554360 s, above 0.4 s; actual wire execution, the full live-handle bound and remote runtime remained unverified. Earlier balanced v3 timing did not show a stable wall-time gain. Neither candidate replaces the hashed graded artifact, and no official score improvement is claimed. [E47]

Where the remaining gap sits

These are observations from public pages, not claims about other entries' methods. In the public Phase 2 board of 2 October (07:04 UTC; displayed values are rounded, and the multiplier is read as displayed score over displayed raw MSE), our entry ranked 26th with raw MSE $2.61e-8$ and multiplier about 0.138. The twelve highest-ranked entries had raw MSE $1.14-2.00e-8$ at multipliers 0.107-0.158, so our cost is inside the leaders' range and the gap is raw accuracy: at our multiplier, a top-five score would need roughly 44% lower raw MSE. Of the 25 entries ranked above ours, 24 report raw MSE below $2.562e-8$, the untruncated closure floor of our method (a matched refit on the uncapped trajectory, mini 80-87, measured in August); the remaining one ranks on a multiplier of 0.101 at raw $2.68e-8$. Our deployed raw MSE is within a few per cent of that floor, so the cap and accounting variants tested here did not demonstrate how to close that historical gap. [E29]

Submission pages also list metered FLOPs, wall time and residual time for all 100 graded networks, including the 50 with sealed scores. For the leading entries inspected, metered FLOPs vary with the network (coefficient of variation 1-3%, about 100 distinct values), and their per-network FLOP vectors correlate 0.85-0.99 across teams; ours vary by 0.05% and correlate -0.78 to -0.91 with theirs. The two largest raw improvements between 27 September and 2 October (-21.2% and -17.3%) coincided with a change from a fixed to a network-dependent FLOP profile, and raw MSE at or below $1.55\text{e-}8$ first appeared on the board between 26 and 30 September. Every page reports the same runtime (flopscope 0.12.1, whetstbench 0.16.1), so these movements are not repricing. Which network property such schedules depend on is not visible from the telemetry. [E29]

One coincidence is worth recording. The reference AUGMENT/SIMPLE closure ratio levels off near 0.43 at widths 512-640, and $0.43 \times 2.562\text{e-}8 = 1.10\text{e-}8$, within 4% of the lowest raw MSE on the board ($1.136\text{e-}8$). The two factors are not strictly commensurable (the 0.43 is an uncorrected reference ratio, the floor includes learned corrections), and under our column cap AUGMENT kept only a 0.89-0.93 advantage (E24). The observation is consistent with leading entries carrying AUGMENT-level information at about 0.11-0.16 of the budget in a representation other than our capped factorized carrier; it does not establish it. [E29]

A second look on 3 October, comparing the public per-network final-layer MSE of three entries with ours on the 50 public networks, sharpened this. Their advantage is nearly uniform across networks (median error ratios 0.89, 0.56 and 0.54 against #333716, with 10-90% ranges of 0.82-1.00, 0.48-0.63 and 0.48-0.62) and only slightly larger where our error is larger (correlation about -0.4). Their per-network FLOP variation is unrelated to their gain or to network difficulty ($|\text{correlation}| \leq 0.18$), so this telemetry does not establish that the observed FLOP variation causes their accuracy gains; Gates C-G above test the candidate explanations we could construct, and none survives beyond a weak, inconsistent gain. One entry in our accuracy class (raw $2.34\text{e-}8$) runs at 26% fewer FLOPs and reports only the final layer. [E33]

Scope of the evidence

These results apply to frozen implementations and evaluated distributions. They do not prove that all cheaper arithmetic programs, all low-rank representations, or all cumulant corrections have been exhausted.

Two tooling cautions apply [E30]. First, Gate B and several earlier diagnostics use a float64 replica that predates the deployed source's fitted pair-delta term (`_PAIR_FINAL_FIT`); against the deployed estimator its predictions differ from layer 2 on by $1.1\text{-}2.6\text{e-}5$ relative, and per-network final MSE by up to about $\pm 2\%$ (1.021, 0.985 and 0.988 on three networks). Replica differences below about 2% are therefore not claims about the deployed estimator; the Gate B conclusions rest on differences of 4% or more. Second, the grader's 100 evaluation networks are not in the public dataset (no name overlap with the mini split and one with the 1,000-network full split, whose weights we do not hold locally), so per-network grader-to-local parity cannot be checked. Only aggregate agreement and the identity of harness and direct execution (bit-identical MSE on the network checked; the harness adds a constant 32,768 FLOPs) were verified.

Finite Monte Carlo references introduce target uncertainty. Development comparisons are vulnerable to selection effects. Independent blocks reduce that risk only before outcomes are observed. Small local timing comparisons do not establish a distribution-wide runtime guarantee. Neither native nor wire diagnostics substitute for official grading. Successfully graded status is also distinct from a completed rule-conformance review. [2]

8. Reproducibility and attribution

The transfer entry #336611 is the tarball

`reports/2026-10-16/e30a-corrector/package/submission_e30c.tar.gz` (SHA-256 5680e97a48030f4b055ff263fc815f81be48d3b5f5a52090119551c66b1dd4c3; `estimator.py` SHA-256 33acb989d10070e2a7e6b18aba30a908d8ad11c40619086dd54e9aa694c56851), built from the public `e30_a` file (SHA-256 6ddd97a96c30397683815c530bfb422f0a49ffdea49e4f7827149ed0a14729b0) by `tools/build.py` in the same folder. On 17 October the organizers moved the evaluators to `flopscope 0.12.3`; #336611, #335981, #335413 and #334793 give identical metered FLOPs and byte-identical predictions under 0.12.1 and 0.12.3 on full 384, the one network checked. This is a saved one-network audit, not an all-network verification. [E48, E49]

The artifact of record is the tarball held by the platform under submission #335413; its SHA-256 is 49a39595b960304d3e3b83384ec8468da12519fbf811d6d886cf3ac8f464df57 and the SHA-256 of the `estimator.py` inside it is 2a3c737090a440fab083587fda218b52d50a382da25ce4acbc3e8c9d621a51e4. The same files are kept in the participant's repository at `reports/2026-10-11/055d-completion/candidates/cand2-v1/` (`estimator.py`, `estimator.json` build proof, `package/submission.tar.gz`), with the submission ledger `reports/2026-10-11/055d-completion/SUBMISSIONS.json`. That source is the frozen #334316 source (archive 99dafcd53bcdff49bb2e6b074f1c270897f9e049f4a0773492e854c9e12408fe, `estimator.py` member ecc25e6b5d26cc4eaa9ae14945344c85db40208f2edf7a6402756cfb915dc1d6, kept inside `reports/2026-10-06/c3bc-submit/submission.tar.gz`) with the corrector tail of Section 6 appended by the frozen builder `reports/2026-10-07/residual-corrector/extractor/deployment/build_candidate.py` (mode trained) from the export `reports/2026-10-11/055d-completion/runs/train-cand2-v1/run/budgets/11400/best/` (`model.json`, `parameters.npz`); the earlier corrector steps #334717 and #334793 are kept under `reports/2026-10-07/overnight-1008/` and `reports/2026-10-08/terminal-followthrough/`. The #334316 source is the #333716 source with the Gate H correction (the lite2 rewrite in `reports/2026-10-05/codex-i/lite2/`), the copy batching of `reports/2026-10-06/codex-j/trackF/sources/estimator_ship.py`, and the coefficient rows and layer condition of the layers 8-14 subset `reports/2026-10-06/codex-j/trackA/native-sources/estimator_candidate3.py`; the four-line difference is reproduced by `reports/2026-10-06/bc-timing/est_c3bc.py`. Its parent #333716 is kept at `reports/2026-10-03/r48b-submit/` (archive 28aeb7b26fc2ff99f2865eb6a2f8289de845deaa189fbb32ea83fefe2cea4062, source dc0663cfe3b2943e7d8452138f275f4f9b9233f67598eab414eb4f58a68318a0), produced from the #333685 source by the generator `reports/2026-10-03/opcut/make_r48b.py`. The earlier artifact describing the operating point of Sections 1-5 is #332177 (archive 8b72d3e1d0e1c32da8acc08007210ec19623608c6691799cb1c2613d9421de41, source 54c8d959713d48e07bb2b1a9eb4552e7cea153dcae1096bdfb18b121ad565af, kept at `reports/2026-09-25/campaign2/c41-query-scratch/submission/submission.tar.gz`); the sibling #332191 is `reports/2026-09-25/campaign2/c49-classical-weight-cache/submission/submission.tar.gz` (archive 3e1d0bd82b9de7ba2ae8b2bd3e4646c4de3c4b4f594cee7d7e1b7903c00183dc, source f487f79f634c769553e8fc5e90d02a5e8465c8f51a0319b121c3bf3f7af66585). All repository-relative paths below refer to the participant's local evidence repository. The frozen source and selected supporting evidence can be provided to organizers; this document does not claim that the repository or large local data caches have already been published. The preregistered protocol, build proof, development pairs, resource pairs, wire audit and official receipt of every candidate C21-C64 are preserved under `reports/2026-09-25/campaign2/<candidate>/`, with the campaign ledger in

[CAMPAIGN2-RESULT-2026-09-25.md](#) and two read-only audits reconciling all five 24 September submission packages, unique source hashes and per-network results. Inference coefficient tables are embedded in the estimator; external training data is not required to load the submission. The response-fitting inputs (per-network native predictions, response matrices, fits and independent optimizer checks) are content-addressed under [reports/2026-09-21/gpu-layerwise/native-fallback/](#) and [reports/2026-09-23/train32/](#). The rank-48 coefficient provenance (six SHA-256 digests of the frozen PLinOpt files) is in [research_assets/campaign_rank48_sept10/provenance.json](#). The historical #331357 release ([whest-starterkit/estimator_review_basis832_20260917.py](#), SHA-256 [090b765dc811b6943ec8fa37494e549b5420db27cc0a3c33880649e0a24cdba5](#)) and earlier snapshots remain separate and must not be substituted for this source.

The Section 4 response fits used only public full-split networks: training full0-7 and full32-63, development full8-15, with full16-31 reserved and never used for fitting or selection. The earlier Section 3 refit used mini-split networks 40-79 for training and mini 80-87 as its development set; mini 80-87 also served as the development set for the arithmetic checks in Sections 5 and 7, and the frozen independent blocks B, D and J are full-split networks 800-831, 864-895 and 672-703. Official scores were never used for fitting.

The later regeneration experiment [E25] reused public full32-63 only for its two matched refits. Its synthetic roles were frozen separately: weight seeds 2026092700-03 for precision design, 2026092704-11 for development and 2026092712-27 reserved for confirmation. Each evaluated network had 32 independent MC blocks of 2^{26} inputs. The development failure left confirmation unevaluated; protected public full16-31 targets were not opened. Its saved-result postmortem generated no new predictions or targets and did not alter the registered assessment. The 2 October gates [E27, E28] used one width-256 reference network with the E24 targets (Gate A) and mini networks 40-59 for fitting with mini 80-87 for diagnostics (Gate B); no protected targets were opened and no submission followed. The 2-3 October submissions [E31, E32] changed no coefficient and opened no protected targets: the batched builds were checked by output bytes against their parents, HFP and ITN by their 30 September-1 October candidate records, and the new measurements are bills, operation counts, handle peaks and official residual and wall times. The Gate H coefficients of #334316 were fitted on mini networks 40-79; mini 80-87 served as development for the layer subset and native port, the official full networks 512-519 were opened once for its fresh confirmation (with 520-527 recorded as exposed), and full 528-535 remain reserved and unopened. The Section 6 corrector used public full networks only: training 608-863 (pretraining), then 32-159 and finally 0-15, 160-383 and 864-999 were added; pretraining development selection on 864-927, then later selection on 384-447; the independent final blocks 256-319, 320-383, 544-607 and 448-463 and 480-527 were each opened once for one candidate's gate and then retired (256-383 were later admitted to the training pool, after their evaluation role had ended); 16-31 and 528-543 stayed protected and 464-479 was excluded as overlapping the public grading set. Official scores were never used for fitting or selection in this lineage either.

The local runtime for the 17 September release and for the 24-25 September campaign was the same: flopscope 0.12.1, whestbench 0.16.1, Python 3.10.11 and NumPy 2.2.6 (the grader pins flopscope 0.12.x and whestbench 0.16.1). The current native and client/server checks used one BLAS thread; some historical tests used four and are labeled separately. The 2-3 October builds used the same versions; their client/server checks ran through [reports/2026-10-03/opcut/rpc_run.sh](#) (separate server and client environments, the 800-array and 4 GiB per-array limits, one BLAS thread), and from that path only output bytes, bills, operation counts and live-array peaks were used, not absolute residual times. The legacy Docker recipe pins Python 3.10.21 and package versions. Local environments do not reproduce the private grader's hardware or complete execution environment.

To regenerate the PDF, use `python tools/build_phase2_final_pdf.py` with ReportLab and Windows Calibri fonts. The canonical source is `writeup-phase2.md`. The historical source and legacy generator are preserved under `docs/history/`; current claims are no longer derived from the chronological Phase 1 manuscript.

AI assistance and responsibility

LLM coding assistants (Anthropic Claude and OpenAI Codex) were used extensively for literature synthesis, mathematical and engineering hypotheses, derivation drafts, estimator and experiment code, log interpretation, preregistered campaign execution, and writing. This consolidated document and PDF tooling were also prepared with their assistance. LLM plausibility is not validation: quantitative claims are tied to recorded experiments or published sources. Explanatory interpretations and limitations are separated from measurements. The participant remains responsible for reviewing and submitting the final account.

References

- [1] Wilson Wu, Victor Lecomte, Michael Winer, George Robinson, Jacob Hilton and Paul Christiano. [Estimating the expected output of wide random MLPs more efficiently than sampling](#), arXiv:2605.05179, 2026. Reference implementation. Cumulant propagation and its Hermite framework are prior work.
- [2] ARC / Alcrowd. [Official Rules, v14](#), inspected 14 September 2026; [Phase 2 announcement](#).
- [3] Oded Schwartz and Noa Vaknin. [Pebbling Game and Alternative Basis for High Performance Matrix Multiplication](#), SIAM Journal on Scientific Computing, DOI:10.1137/22M1502719. The multiplication identity and alternative-basis method are prior work.
- [4] Jean-Guillaume Dumas, Clément Pernet, Alexandre Sedoglavic and Petr Tichavský. [arXiv:2609.12027](#), 2026: a non-commutative algorithm multiplying 4×4 matrices with 48 multiplications; coefficients from the PLinOpt repository (`4x4x4_48_canonical_{L,R,P}`). The rank-48 scheme is prior work; only its selective placement inside the metered interface and its verification are ours.

Local evidence index (repository-relative paths)

- E1: `whest-starterkit/logs_2026-09-09/p2vp_official/official_graded_promotion.json` and its recorded source/archive digests.
- E2: `writeup-phase2-addendum-2026-09-07.md` (mixed pair repair, covariance-shaped K4, independent blocks B and D).
- E3: `SELECTIVE-SMT-RESULT-2026-09-14.md` and `reports/2026-09-14/xa-selective-views16/` (independent J).
- E4: `INCUMBENT-JOINT-COMPILER-2026-09-14.md` and `XD-FINAL-REUSE-VALIDATION-2026-09-14.md` (calls, timing and reuse).
- E5: `writeup-phase2-addendum-2026-09-10.md` (56-parameter surrogate-response test).
- E6: `writeup-phase2-addendum-2026-09-12.md` (nonterminal output-row policy).
- E7: `docs/history/writeup-phase2-through-2026-09-12.md` (historical diagnostic context).
- E8: `REVIEW-FOLLOWTHROUGH-2026-09-15.md`, `SOURCE-K4-OFFICIAL-RESULTS-2026-09-17.md` and the frozen native-refit manifests (literal trajectory, Wick discrepancy, current tables).
- E9: `NONNEGATIVE-FINAL-RESULTS-2026-09-17.md`, `XA-CURRENT-RESULTS-2026-09-17.md`, `XD-CURRENT-RESULTS-2026-09-17.md` and their `reports/2026-09-17/daytime/*/official/final_summary.json` and source/archive receipts (September 17 official evidence).
- E10: `BASIS832-RESULTS-2026-09-17.md`, `BASIS-FINITE-COST-2026-09-17.md`, frozen integration protocol and `reports/2026-09-17/daytime/basis832/official/` (implemented recurrence, numerical checks, local failure and official

receipt).

E11: K22-FINAL-LOSS-RESULT-2026-09-19.md, OFFICIAL-SIX-RESULTS-2026-09-21.md, NINE-OFFICIAL-RESULTS-2026-09-21.md (K22 and K4 final-loss fits, #331604, #331650, #331740).

E12: GPU-LAYERWISE-RESULT-2026-09-21.md, NATIVE-LAYERWISE-PROTOCOL-2026-09-21.md, NATIVE-LAYERWISE-RESULT-2026-09-21.md, T4-EXPLORATORY-SUBMISSION-RESULT-2026-09-22.md (GPU transfer gate, 42-coordinate native fit, #331836).

E13: NINE-OFFICIAL-RESULTS-2026-09-23.md, T5-CAP7168-RESULT-2026-09-22.md, TRAIN32-RESULT-2026-09-23.md (sibling refits and #331910, cap-7168 refit, 32-network fit).

E14: COMPETITOR-LITERATURE-SCAN-2026-09-18.md, reports/2026-09-19/g1_k4core_rank/, RANK-STRATEGY-2026-09-21.md (low-rank core, pair fourth-order oracle ceiling, wall-time kernels).

E15: SUBMISSION-CAMPAIGN-2026-09-18.md, FOLLOWUP-RESULTS-2026-09-18.md, TOMORROW-SUBMISSION-DIRECTION-2026-09-19.md (padding/sharing, compact basis, schedule extension).

E16: CAMPAIGN-RESULT-2026-09-24.md, CAMPAIGN2-RESULT-2026-09-25.md, reports/2026-09-24/campaign/rank48-*/, reports/2026-09-25/campaign2/ (rank-48 component gates, selective placement, dtype finding, 44-candidate ledger, residual forecasts and official receipts for #332154, #332159, #332177, #332191).

E17: TRAIN32-OFFICIAL-RESULT-2026-09-23.md, RANK10-OVERNIGHT-RESULT-2026-09-24.md, reports/2026-09-24/campaign/gn/ (32-network refit #332017, 200-network replica fits, relinearization, output offset/gain diagnostics).

E18: V29-HYBRID-G0-2026-09-24.md, reports/2026-09-24/campaign/v29/, reports/2026-09-24/campaign/ensemble/, reports/2026-09-24/campaign/plateau/, reports/2026-09-24/campaign/INCUMBENT-832-FRINGE-REVIEW.md (hybrids with the public entrant estimator, plateau pruning, fringe review).

E19: research-note.md sections 9cj-9cl (gate-dead-neuron pruning: threshold, bill and raw effect, graded #329622/#329636) and 9bh-9bi with whest-starterkit/_d_acc_meanfit.py (third-cumulant channel on the untruncated versus capped trajectory; #328840).

E20: reports/2026-09-25/score-s1/ (cap 8,192 and 8,704 on the deployed arithmetic) and reports/2026-09-26/score-s3/ (retained-coordinate relaxation), both with per-network development receipts.

E21: reports/2026-09-26/augment-ladder/ (RESULT.md, g0-cost.json, g1a-correct-orientation/): symbolic cost bound of the regenerated-core formulation and the eight-network OFF/ON re-run of the donor regeneration law.

E22: reports/2026-09-26/floor-cap/ (bill-versus-cap table on the deployed arithmetic, matched refits at caps 4,096 and 4,608).

E23: reports/2026-09-26/student/ (data admission audit, FLOP budgets of both students, e1/ recurrent-student history, e1-a/ covariance-chain student result).

E24: reports/2026-09-27/augment-trunc/ (AUGMENT versus simple kind under four truncation policies at width 256, depth 16; B2 reproduction; noise floors).

E25: reports/2026-09-27/regen-shared-v2/PRODUCTION-PROTOCOL.md, evaluation/PROTOCOL.md, evaluation/dev/result.json, evaluation/dev-completion-audit-20260929/ and matched-refit/selection.json within that directory (frozen roles and gates, completed development failure, prediction integrity and training refits); reports/2026-09-29/regen-postmortem/RESULT.md, analysis.json and source-audit.md within that directory (saved-result decomposition, input hashes and component scope).

E26: reports/2026-10-02/official/333031/ and reports/2026-10-02/official/333203/ (summary.json, page-evaluation.json: official score, raw MSE, per-network FLOPs, wall and residual); reports/2026-09-29/overnight-until0700/FINAL-REPORT.md and lane-h-submission/graded-review-333031.md within

that directory (B60 composition, fresh eight-network evaluation, official review; source SHA-256 `cb014fc34d5c11647c76cbc58c0ffe3ac947bd39e287e17aa9474b552e897c38`); `reports/2026-09-30/overnight-until0700/FINAL-REPORT.md` and `CANDIDATE-LEDGER.md` within that directory (SFP1 byte identity and per-network FLOP saving; source SHA-256 `62a01f5b89b93354650f6bd286f908a72ec550aed187d232197f999981cc6674`).

E27: `reports/2026-10-02/research-direction/PROTOCOL-GATES-2026-10-02.md` (frozen gate thresholds, sections 2-3; verdicts, section 11) and `reports/2026-10-03/lifetime-trunc/ (RESULT-DRAFT.md, freeze.json, audit.json, cost.json; Gate A)`.

E28: `reports/2026-10-03/lambda-meanfit/ (SCAN.md, scan.json, b1b-fit.json, ablation-*.json; Gate B scan, per-layer fit and path exclusions)`.

E29: `reports/2026-10-02/research-direction/RESEARCH-DIRECTION-2026-10-02.md` and `public-board-20261002T070432Z.json` in that directory (board snapshot); the forensic report `reports/ARC*.md` (single file, Korean title) with its per-network telemetry under the matching `research_notes/` folder; `research-note.md` section 9bm (AUGMENT/SIMPLE ratio versus width and the $2.562e-8$ untruncated floor).

E30: `reports/2026-10-03/tooling-audit/CHAIN-TOOLS-AUDIT.md` and `reports/2026-10-03/tooling-audit/ref-runtime/REF-RUNTIME-TOOLS-AUDIT.md` (replica-to-deployed gap, grader evaluation networks absent from public data, harness identity).

E31: `reports/2026-10-03/itn1-submit/RESULT.md` (#333546 and the partly withdrawn forecast reading), `reports/2026-10-03/HFP-HBR-RESULT.md` (#333552, #333553), `reports/2026-10-03/overnight/REPORT.md` (#333561, residual medians and maxima across all graded runs, operation census, stop decision), each submission folder's `harvest/` (summary.json, SUMMARY.md, page-evaluation.json) including `reports/2026-10-03/hfpitn/harvest/`; HFP and ITN candidate records in `reports/2026-09-30/overnight-until0700/CANDIDATE-LEDGER.md` and `reports/2026-10-01/overnight-until0700/ (CANDIDATE-LEDGER.md, FINAL-REPORT.md; local forecasts 0.4237, 0.4239 and 0.4089 s)`; `reports/2026-10-03/tooling-audit/ref-runtime/r6-sfp1-bill-mlp80.json` (28,018 operations for #333203).

E32: `reports/2026-10-03/bhfp-submit/RESULT.md`, `reports/2026-10-03/b2hfp-submit/RESULT.md` and `reports/2026-10-03/r48b-submit/RESULT.md` (#333634, #333685, #333716: operation counts, residual and wall, paired differences, arena sizes), with `submission.tar.gz`, `estimator.py`, `receipt.json` and `harvest/` in each folder; generators `reports/2026-10-03/opcut/make_bslic.py` (batched rank-40 slice pairs) and `make_r48b.py` (batched rank-48 query pair); the local client/server harness `rpc_run.sh` and its logs under `rpc/` in the same folder (live-array peaks 788 before batching, 794 after, 800 with the 50-view probe `probe_views50.py`); census scripts `sites.py`, `swpaths.py`, `fcensus.py`, `slice21share.py`, `paired_residual.py` and their outputs in `census-and-paired.txt` (slice21 share of operations, Strassen copy share of the bill, per-network paired residual differences); `reports/2026-09-25/campaign2/c44-full-tile-dedicated/RESULT.md` (C44 resource forecasts).

E33: `reports/2026-10-03/augment-cost-bound/` (Gate C), `reports/2026-10-03/plane-k3/` and `tools/plane_k3_20261003.py` (Gate D), `reports/2026-10-03/adaptive-cap/` and `tools/adaptive_cap_20261003.py` (Gate E), `reports/2026-10-03/extra-legs/` and `tools/extra_legs_20261003.py` (Gate F), each with `PROTOCOL.md` committed before its first run; `reports/2026-10-03/opcut/FULL48-NEGATIVE.md`, `reports/2026-10-03/opcut/LEVERS-RECHECK.md` (rechecks); `reports/2026-10-03/recon/RECON-2026-10-03.md` with the harvested pages of #333581, #333587 and #333663 (public telemetry comparison).

E34: `reports/2026-10-03/legs-polar/` and `tools/legs_polar_20261003.py` (Gate F2: symmetric-column legs, seeds 0-7); `reports/2026-10-04/leg21-replica/` (`PROTOCOL.md` committed before its runs, `RESULT.md`, driver `run.py`, per-arm JSON), `reports/2026-10-04/leg21-lifetime/` (lifetime-k free F1, LEG21=4), `reports/2026-10-04/leg21-xproj/` (X projected with F1 exact, LEG21=5; `PROTOCOL.md` committed before its runs) and the replica knob LEG21 in `whest-starterkit/_d_acc_replica.py` (default off; validations to $1e-12$).

E35: `reports/2026-10-04/iso-feed-cost/NOTE.md` (cost arithmetic of the (2,1,1) feed) and `reports/2026-10-04/iso-feed-check/` (Gate G: `PROTOCOL.md` committed before its runs, `run.py`, `analyze.py`, `summary.json`, `RESULT.md`; replica knobs FEED, FEED_GAIN, FEED_SHAPE, default off); the 14 September matched refit is

in [reports/2026-09-14/claude-audit/](#).

E36: [reports/2026-10-05/codex-g/](#) (Gate H: [FAILURE-MAP.md](#) and [CANDIDATES.md](#) for the selection, [PROTOCOL.md](#) with amendments 1-3 committed before their runs, [HOLDOUT.md](#) for the reserved fresh networks full 512-519, [PILOT-RESULT.md](#) (step-size stop), [PILOT-RESULT-2.md](#) (selection-frozen pilot), [MAIN-RESULT.md](#) (forty-network fit, out-of-fold and held-out screen), run directories with launch records and per-network JSON); replica knobs [U01](#) and [U01_SELECTION](#) in [whest-starterkit/_d_acc_replica.py](#) (default off). The September predecessor is research-note [\\$9bs\(10\)](#). Native stage: [reports/2026-10-05/codex-g/](#) [PROTOCOL-AMEND-4.md](#) to [-7.md](#), [NATIVE-RESULT.md](#), [NATIVE-DEV-RESULT.md](#), [FRESH-RESULT.md](#), [-2.md](#), [-3.md](#) (fresh full 512-519, final port [native/estimator_v3.py](#)); submission #334003 and per-network grader telemetry [reports/2026-10-05/u01v3-submit/](#) (with #333716 for comparison); wall-margin variants [reports/2026-10-05/u01-wallsafe/RESULT.md](#). Cheaper rewrites: [reports/2026-10-05/codex-i/](#) ([DESIGN.md](#), [PROTOCOL.md](#) with amendments 1-4, [LINE-REVIEW.md](#), [DIAG-LITE.md](#), [RESULT-3.md](#), [CALLS.md](#), [RESULT-LITE2.md](#)); the quad-tree layout alternative that was closed at its design stage is [reports/2026-10-05/codex-h/](#). Layer subsets, copy batching and the paired timing: [reports/2026-10-06/codex-j/SUMMARY.md](#) and [reports/2026-10-06/bc-timing/](#) ([PROTOCOL.md](#) committed before the runs, [RESULT.md](#)). Submission #334316 and its per-network grader telemetry against #333716: [reports/2026-10-06/c3bc-submit/](#) ([README.md](#), [harvest/SUMMARY.md](#)).

E37: [reports/2026-10-07/overnight-1008/FINAL-REPORT.md](#) and [reports/2026-10-07/residual-corrector/](#) (collection of 320 networks, extractor [extractor/extract.py](#) and feature schema, convergence protocols, B600 development and independent final receipts, #334717); [reports/2026-10-08/terminal-followthrough/FINAL-REPORT.md](#) and [continuation-1300/](#) (terminal objective, 28,800-update run, DEV64 and final 320-383 receipts, #334793); [reports/2026-10-08/expanded384-followthrough/](#) (pilot protocol, 384-network run, replication over three seeds, the 055d candidate, A/A ENV_HOLD under load).

E38: [reports/2026-10-11/055d-completion/](#) ([PROTOCOL.md](#) with the candidate-2 preregistration, [STATE.json](#), [SUBMISSIONS.json](#); the lean evaluation harness [fwd.py/batch.py/score.py](#) whose scorer reproduces the earlier saved DEV64 statistics bit-exactly and whose fresh incumbent forward is byte-identical to the saved output; [collect_lean.py](#) (bit-identical to the original collector on network 384); [train_lean.py](#) (bit-exact replay of the pilot trajectory on the 384 pool); runs [dev64-candidate-v1](#), [final64-v1](#) (the 055d sealed final, 37/64), [train-cand2-v1](#), [dev64-cand2-v1](#), [final2-cand2-v1](#) with [SCORED-*.json](#)); [reports/2026-10-09/score-recovery/](#) (width 128, second message channel, fixed readout refit, Signed8 extra state, auxiliary-versus-terminal comparison, all negative).

E39: [reports/2026-10-11/055d-path-recovery/](#) (numerical path diagnosis: NumPy contraction order versus the metered pairwise path, adapter byte-equality on mini80 and DEV384/400/416/432; the quiet-host timing harness, its four first-execution amendments [TIMING-HARNESS-AMENDMENT-v1.md](#), and the passing A/A and candidate blocks [results/timing-aa-v5/](#), [results/timing-cand-v5/](#) with per-leg receipts).

E40: [reports/2026-10-12/g2-pure-feed/](#) ([PROTOCOL.md](#) with amendment 1 registered before the runs, [run_g2.py](#), [arms/*.json](#), [ANALYSIS.json](#), [RESULT.md](#): the (2,1,1) feed on the pure chain, capped and untruncated); [reports/2026-10-11/055d-completion/runs/train-cand3L-v1](#), [train-cand3D-v1](#) and [train-decay-validate-v1](#) (horizon and learning-rate continuation runs; the thin loop's bit-exact replay of the frozen trainer at update 96).

E41: [reports/2026-10-11/055d-completion/runs/train-cand4U-v1](#) (per-layer unshared corrector weights: [train_unshared.py](#), [INITIAL.json](#) with the step-0 parity to candidate 2, [evaluations/update-*.json](#) for every epoch, [RESULT.json](#)), the preregistration and result in [reports/2026-10-11/055d-completion/PROTOCOL.md](#) (Candidate 4, Amendment 1), and the prepared but unused deployment port [build_cand4.py](#) / [check_cand4.py](#) (parity on invented inputs only).

E42: [reports/2026-10-12/confine-gate/](#) ([PROTOCOL.md](#) with Amendment 1 registered before any refit result, [run_gate.py](#), [analyze_gate.py](#), [ANALYSIS.json](#), [RESULT.md](#), [arms/*.json](#) with per-arm refit tables and held-out predictions, [smoke/*](#) with the thread-matched bit-identity pair); replica knobs [CONFINES](#), [CONFINES_MODE](#), [CONFINES_W](#) in [whest-starterkit/_d_acc_replica.py](#) (default off).

E43: [reports/2026-10-12/regen-rebuild/](#) ([PLAN.md](#) written before the runs and extended with the Stage 2 preregistration before its refits, [RESULT.md](#), [stage1_compare.py](#), [stage1b_ablate.py](#), [stage1/*.json](#), [stage2/*.json](#) with per-round refit

tables and held-out predictions for the two Stage 2 arms, run with `../confine-gate/run_gate.py`; `stage2_lamcheck.py` and `stage2_recheck/*.json` for the λ -table and identity-table ablations; the lean chain of 504aldo/whest-p2-cumulant-k3 run against ARC reference snapshot a359031 on our held-out networks 80–87 and on mini networks 0–1 for the reproduction check; logs in the session scratchpad).

E44: `reports/2026-10-12/checkpoint-average/RESULT.md`, `evaluate-v1/output/RESULT.json`, `audit-v2/output/RESULT.json`; `reports/2026-10-12/context-head/RESULT.md`; `reports/2026-10-12/external512/PILOT-RESULT.json`, `PILOT-TERMINAL-AUDIT.json`, `FINAL-REPORT.md` (fixed averaging, conditional context calibration and the 512-network cost gate).

E45: `reports/2026-10-13/bidirectional-cost/RESULT.md`; `reports/2026-10-13/cost-aware/FINAL-REPORT.md`; `reports/2026-10-13/autonomous-research/STATE.json` and its `CONDITIONAL-RESULT.json`, `GEOMETRY-AUDIT-v1.json`, `REUSE32-RESULT.md`, `ACCUM-RESULT.md`, `SAM-RESULT.md`, `SP-RESULT.md`, `LBFSGS-RESULT.md`, `LORA-RESULT.md`, `DROPOUT-RESULT.md`, `SCV-FINISH-RESULT.md` (separate fixed studies and saved-result audits; no new official grade).

E46: `reports/2026-10-14/first-layer-cv/PROTOCOL.md`, `MEASURE-RESULT.json`, `AUDIT-RESULT.json`, `RESULT.md`; `reports/2026-10-13/math-only-review/OVERNIGHT-FINAL-REPORT.md` and `NIGHT-MATH-26-APPLICABILITY-REVIEW.md` (fixed control-variate screen and mathematical scope; no cheap-construction or score theorem).

E47: `reports/2026-10-14/codex-v4-validation/RESULT.md`, `RESIDUAL-DIAGNOSIS.md` and the original run receipts named there (native-only unsubmitted candidates; wire, handle and remote-runtime qualifications incomplete).

E48: Paths in E48-E50 are relative to `reports/2026-10-16/e30a-corrector/` unless written in full. `PROTOCOL.md` (with amendments), `SPLIT.json`, `RESULT.md`, `SUBMISSIONS.json`, `runs/full-v1/` (`log.jsonl`, `BEST.json`, `DEV-GATE.json`, `FINAL-GATE.json`), `deploy-check/GATE3.json`, `resource/final-r3/RESOURCE-GATE.json`, `smoke/e30a-full-00384.json`, `deploy-check/e30c-full-00384.json`, `upstream/README.md`, `upstream/LICENSE`, `package/src/estimator.py`, `tools/` (transfer to `e30_a`, #336611); organizer clarification, forum 18240.

E49: `reports/2026-10-16/e30a-corrector/PROTOCOL-ENSEMBLE.md`, `runs/ENSEMBLE-HOLDOUT-GATE.json`, `deploy-check/GATE3-ENS.json`, `PROTOCOL-GRU.md`, `runs/gru-v1/`; `reports/2026-10-17/flopscope-0123-audit/RESULT.md` (ensemble and GRU follow-ups; evaluator-version audit).

E50: `reports/2026-10-16/e30a-corrector/PROTOCOL-DATA.md` (validation record, decision and amendment), `runs/lc348-B/`, `runs/lc348-C/`, `runs/gruS560/`, `synth/synth.log`, `tools/mc_truth.py`, `tools/synth.py` (learning curve and synthetic-data scaling; synthetic labels and weights are kept locally, not in the repository).