Private-Safe Structural Monte Carlo for Wide Random ReLU MLPs

Phase 1 Algorithmic Contribution — Submission 326960

Exact symmetry identities, conservative activation pruning, and residual-aware matrix multiplication

LLM usage disclosure

LLMs were used extensively for ideation, implementation, analysis, code review, and drafting. I directed the research, selected experiments, reviewed the evidence, and made the final decisions. LLM assistance contributed to much of the experimental and submission code, the interpretation of measurements, and this write-up. I do not claim that every idea or line of code was independently produced by me. Numerical claims below were checked against the committed source, experiment logs, validation artifacts, and official grading results. Interpretations are identified as interpretations rather than proofs.

Phase 1 submission	Official adjusted score	Status
326960	`1.727503070366728e-7`	Graded successfully; 50/50 completed, 0 failures

The report documents the promoted source `estimator_sweep_dim64.py`. The official result above is the confirmed score; no unrecorded final-layer MSE is claimed.

The short version

The task is to estimate the layerwise expectations of a fixed, wide, deep, bias-free ReLU MLP under Gaussian input. The estimator remains Monte Carlo, but both its samples and its execution are adapted to exact structure in the problem and in the realized MLP:

1. A reproducible Gaussian cloud is empirically centered and whitened.
2. Antithetic points `x` and `-x` reduce odd sampling error.
3. The first-layer antithetic computation uses the exact identity `ReLU(-z) = ReLU(z) - z`, avoiding a duplicate product.
4. A lead block identifies rows that are never active; those rows are removed from later products.
5. Stable final-layer columns are treated as exactly zero or linear; uncertain kink columns retain full sampling.
6. Large aligned products use a two-level Strassen/Winograd recursion, while smaller products use ordinary matmul to avoid residual Python overhead.
7. The cost model assumes the worst case—no pruning—so the production path remains budget-safe.

The principal lesson is that variance reduction and cost reduction must be designed together. Exact identities remove arithmetic without changing the estimator, while conservative certificates remove only contributions known to be zero or linear.

1. Problem and estimator

For weight matrices `W_0, ..., W_{L-1}`, the network is

```
h_0 = ReLU(x W_0)
h_l = ReLU(h_{l-1} W_l),    l = 1, ..., L-1.
```

The estimator returns an estimate of `E[h_l]` for every layer. Deep ReLU compositions are difficult to integrate directly because activation gates change across layers, dead coordinates create irregular sparsity, and a small number of high-variance MLPs dominate aggregate error.

The challenge score also charges both instrumented FLOPs and residual wall time:

```
effective cost = instrumented FLOPs + lambda * residual seconds
lambda = 1e11 FLOP/s.
```

Consequently, a nominally cheaper arithmetic identity can lose if it introduces too many small Python-level operations.

1.1 Moment-matched Gaussian cloud

For each MLP, the program creates a deterministic Gaussian sample cloud from `mlp.seed`. It centers and whitens the realized cloud, then fuses the whitening into the first weight product. This removes avoidable first- and second-order sampling imbalance without moving numerical work outside the official `flopscope.numpy` interface.

1.2 Antithetic pairing and exact first-layer sharing

The sample cloud contains paired points `x` and `-x`. For the first preactivation `z = x W_0`,

```
ReLU(-z) = ReLU(z) - z.
```

Thus one half of the first-layer product supplies both members of every pair. This is value-preserving apart from ordinary float32 operation-order effects. Later layers are still evaluated for both members; the identity applies specifically to the first layer.

The same pairing also halves the empirical Gram computation:

```
[x; -x]^T [x; -x] = 2 x^T x.
```

1.3 Pilot-frozen structural pruning

A lead block of 1,024 samples is propagated at full width. For the tail, a row is retained only if it fired at least once in the lead block. Because post-ReLU activations are exactly zero on removed rows, the corresponding terms in the next matrix product are exactly zero. Indices are conservatively rounded to an alignment boundary needed by the fast product.

This mask is frozen before the production samples are processed. It is therefore predictable, reproducible, and compatible with a worst-case cost bound. The implementation does not use MLP names, public instance

identities, or answer tables.

1.4 Terminal classification

For the scored layer, the lead block estimates each output column's sign margin. A column with a conservative positive margin is evaluated linearly; a column with a conservative negative margin is set to zero; all remaining columns are classified as kink columns and receive the full nonlinear sampled computation.

For a stable-on column, the sample mean commutes with the final linear map, so folding it is algebraically exact under the classification certificate. The uncertain class is deliberately retained rather than aggressively approximated.

1.5 Production configuration

The selected source uses the following fixed configuration for the width-256, depth-32 challenge regime:

Parameter	Selected value
Dense-to-masked split	layer 6
Full-width lead block	1,024 samples
Minimum lead firing count	1
Terminal classification margin	1.0 column standard deviation
Strassen/Winograd recursion	2 levels
Minimum recursive matrix dimension	64
Minimum sample dimension for recursion	8,192
Antithetic first-layer reuse	enabled
Symmetry-aware Gram computation	enabled

The mask is rounded conservatively to the alignment required by the recursive product. The selected build does not use the experimental Cholesky whitening path; that path was measured separately and remains disabled in the submitted source.

2. Counted computational identities

2.1 Symmetry-aware Gram

The empirical Gram matrix is symmetric. The submitted implementation expresses it through the symmetry-aware `flopscope.numpy` operation so that only distinct entries are charged. Research validation compared the result with an ordinary product and found identical values. This is a cost-model reduction of the same computation, not a precision approximation.

2.2 Two-level Strassen/Winograd products

Large, aligned matrix products use an exact two-level Strassen/Winograd recursion. The recursion reduces counted scalar multiplications but introduces additions and Python dispatch. A minimum-dimension and

minimum-sample cutoff therefore selects it only where it pays. Small, odd, and narrow products fall back to native matmul.

A level-three recursion reduced nominal FLOPs further but increased residual runtime enough to lose in paired tests. It was not shipped.

2.3 Dtype, accounting, and fallbacks

The bulk numerical path is float32. Copies, gathers, concatenations, elementwise operations, and recursive additions remain inside the official metered namespace. The estimator does not use a custom native numerical kernel or an unmetered compute path.

The sample count is sized against a conservative no-pruning cost model. Numerical safeguards retain a finite analytic fallback if whitening or the sampled path encounters trouble.

2.4 Exact operations versus conservative decisions

The estimator separates value-preserving identities from decisions that trade a certified risk for lower cost:

- **Exact identities:** antithetic first-layer reuse, the factor-of-two antithetic Gram relation, symmetry-aware Gram assembly, and linear sample-mean folding.
- **Conservative structural decisions:** the lead-block dead-row mask and terminal stable-on/stable-off classification. These decisions use margins and retain uncertain units rather than forcing a classification.
- **Runtime decisions:** Strassen eligibility, recursion depth, block size, and alignment. These do not alter the intended estimator; they determine whether counted FLOP savings outweigh residual dispatch time.

3. What the submitted program does

1. Build the deterministic centered/whitened antithetic cloud.
2. Run the lead block at full width and freeze the dead-row mask.
3. Run the remaining samples through dense early layers and masked tail layers.
4. Reuse the exact first-layer antithetic identity.
5. Apply the terminal stable-on/stable-off/kink classification.
6. Return analytic or sampled layer means according to the layer and certified path.
7. Check finiteness and use the fallback ladder if necessary.

The high-count sampled computation is concentrated where it matters most: the terminal output and the uncertain paths leading to it. Earlier layers still provide the realized activation structure used by the estimator.

4. Evidence and ablations

The following results concern our own estimator family and selected source; they are not claims about unrelated submissions.

Checkpoint	Official adjusted score	Main change	Decision
Private-safe two-layer baseline, #326558	`2.912248982e-7`	Antithetic/whitened baseline	Superseded
Structural sweep, #326917	`1.727982882e-7`	Publicly documented structural pruning and terminal classification, independently integrated and validated	Promoted temporarily
Cutoff-64 derivative, #326960	`1.727503071e-7`	Residual-aware Strassen cutoff `48 -> 64`	Selected
Cutoff-80 derivative, #326963	`1.729227170e-7`	More conservative recursion cutoff	Rejected after official grading

The cutoff-80 result is an important negative result: a lower local cost proxy did not transfer to a better official score. It reinforced that residual-time behavior and private-distribution validation matter more than a FLOP-only screen.

Additional internal ablations were:

Internal variant	Result	Decision
Strassen recursion level 3	Lower counted FLOPs, but substantially higher residual runtime	Rejected
Lead block 2,048 instead of 1,024	No compensating accuracy gain	Rejected
Final margin 0.5 instead of 1.0	Negligible measured benefit and weaker conservatism	Rejected
Earlier structural split	Small cost saving but worse held-out accuracy	Rejected
Minimum firing count 2	Small-screen promise but pilot-mask bias on larger validation	Rejected
Experimental Cholesky whitening	Distributionally plausible and cheaper, but not selected for the submitted build	Not shipped

These tests separate statistical changes from implementation changes. The selected cutoff-64 build was chosen because it improved the cost/runtime tradeoff without changing the estimator's structural decisions.

Other exploratory directions—including control variates, higher-order analytic closures, and alternative sampling tables—were not promoted when paired validation showed worse error, bias, runtime, or insufficient private-safe evidence.

5. Limitations

This is still a sampling-heavy estimator. Structural pruning lowers the price of the samples but does not eliminate the rare, high-variance MLP tail. The terminal classification is conservative but depends on a finite pilot and can therefore trade a small amount of bias against cost. Strassen is beneficial only in sufficiently large blocks; using it indiscriminately is counterproductive because residual Python time is part of the score.

The estimator is specialized to bias-free ReLU networks and to the width/depth regime of the challenge. The harmonic explanation for why antithetic sampling and empirical whitening help is plausible and supported by measurements, but is not presented as a complete theorem for the depth-32 integrand.

6. Claimed contribution

The contribution is narrower than a complete solution to mechanistic expectation estimation:

- exact ReLU and antithetic identities remove duplicated work;
- empirical whitening and antithetic pairing reduce avoidable sampling imbalance;
- pilot-derived activation certificates expose a safely sparse tail;
- terminal linearity is used only for certified stable columns;
- residual-aware fast matrix multiplication reduces counted work without unmetered computation; and
- conservative worst-case sizing keeps the estimator safe on pathological MLPs.

Together these choices produced the confirmed private-safe Phase 1 submission **326960**, with official adjusted score **`1.727503070366728e-7`** and zero graded failures. The broader lesson is that statistical design, activation structure, exact algebra, and evaluator-aware systems work can reinforce one another—but only when each claimed gain is checked on disjoint or official evidence.