Sparse MoE Inference on Four Arm Cores: Coordination, Memory Lifecycles, and Failed Optimizations

Tanvir Ahmed
github.com/mailtotanvir/zincrom
Zenodo Concept DOI: 10.5281/zenodo.22741249

September 2026

Abstract

Sparse mixture-of-experts (MoE) models activate only a subset of their expert parameters per token, but logical sparsity does not automatically become bounded physical memory or predictable latency. This report evaluates Zincrom, a split execution architecture that uses Erlang/OTP (BEAM) for supervision and lifecycle coordination while native C/C++ and `ggml` paths perform routing and selected-expert tensor computation. All primary measurements were made on one Oracle OCI A1 instance with four Arm Neoverse-N1 cores and 24 GiB RAM using OLMoE (1B active / 7B total, Q4_K_M). In a 1,000-iteration isolated harness, BEAM coordination measured 74 μs at p99, 3.36% of selected-expert compute at 2,205.553 μs p99. The split path matched a pinned `llama.cpp` reference exactly across three single tokens and causal batches 2, 4, 8, and 16 at all 16 layers; batch 32 retained exact expert selections but missed the strict output tolerance at layer 2. Linux fault-around inflated a theoretical 12.5% expert working set to 27.7% page presence; `MADV_RANDOM` restored 12.88%, and synchronized layer-completion eviction held concurrent expert residency to 3.53–6.11% with 85.4–90.8% throughput retention in tested schedules. Perfect-knowledge local prefetch and Redpanda delivery of 31.1 MiB expert shards both failed their latency gates. The resulting contribution is a bounded systems result: inexpensive coordination is feasible, but memory lifetime, admission, and transport policy must be measured explicitly for each host and model.

1 Introduction

Sparse MoE models offer an attractive systems bargain: OLMoE routes eight of 64 experts for each token, so the logical active fraction of expert parameters is 12.5%. On a small CPU host, that suggests a path to running a model whose total parameter footprint exceeds the practical working set. The bargain is incomplete. File-backed mappings follow operating-system paging policy, concurrent tokens touch unions of experts, request lifetimes overlap, and the data needed by a future expert must arrive before its layer executes. Sparsity is therefore a scheduling and memory-lifecycle problem as much as a model property.

Zincrom separates those responsibilities. BEAM processes supervise requests, admission, barriers, and page lifetimes. Native code retains the numerical kernels and model representation. This report asks whether that split is both correct and useful on a deliberately constrained platform, and records failed gates alongside successful ones.

1.1 Contributions

The completed empirical phase makes five narrow contributions:

1. **Verified split execution.** The OLMoE router and selected-expert path are separated while preserving the pinned reference's expert IDs, weights, and outputs across every layer for the supported single-token and causal-batch cases.
2. **Measured coordination cost.** BEAM dispatch and confirmation consume 3.36% of selected-expert p99 compute in the isolated 1,000-trial harness; the worker processes perform no weight I/O or expert arithmetic.
3. **Explicit physical-memory lifecycle.** Access advice suppresses fault-around, and a layer-completion barrier bounds concurrent expert residency rather than allowing routed pages to accumulate across token and request lifetimes.
4. **Host-bounded admission and cache policy.** Four active decodes satisfy the tested tail-latency bound; aging size-aware queuing repairs a deep-queue fairness failure; routed-expert cache capacity differs across OLMoE, Granite, and Qwen.

5. **Useful invalidations.** Local lookahead fails even with oracle future routes, Redpanda shard delivery misses the measured compute window, and the historical 8B Operational Efficiency Factor (OEF) collapse is superseded after correcting its denominator.

1.2 Empirical envelope

All performance conclusions are restricted to one Oracle OCI A1 virtual machine: four 64-bit Arm Neoverse-N1 cores, 24 GiB physical RAM, one NUMA node, and no swap. The reference lineage is `llama.cpp` commit `6e9007ae6`. Primary split-path measurements use a Q4_K_M OLMoE-1B-7B model. Granite-3.1-3B-A800M and Qwen1.5-MoE-A2.7B-Chat appear only in the explicitly identified policy-portability experiments. The evidence cutoff is 2026-07-25. Runtime 2.0, FFT semantic routing, and the proposed cognitive core are outside the empirical scope.

2 Architecture and Experimental Method

2.1 The split boundary

Figure 1 shows the control/execution boundary. The native path loads the GGUF tensors, evaluates the gate, selects top-K experts, and executes their projections. BEAM supervises the request and expert-worker lifecycle, coordinates completion, and decides when a layer's expert pages can be released. The boundary is intentionally narrow: the measured workers acknowledge dispatch and completion; they do not move weights through the Erlang heap or perform GEMM there.

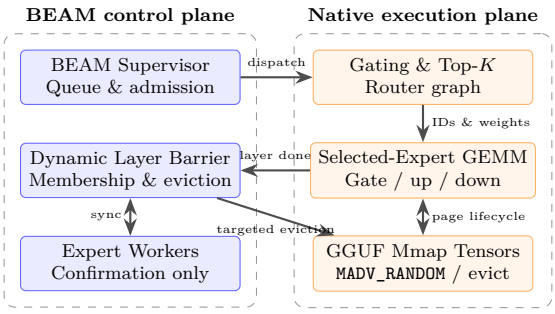

Figure 1: Boundary between BEAM control and native execution. BEAM manages admission, completion barriers, and eviction triggers; native routines route and execute selected experts directly from mapped GGUF storage.

For a layer, the native router evaluates the gating graph and materializes expert IDs and weights. The selected-expert path then evaluates gate, up, and down projections for those experts. Across concurrent contexts, the supervisor waits until every active member has completed the layer before applying targeted eviction. Dynamic membership changes occur between token rounds so shorter requests and cancellation do not deadlock the barrier.

2.2 The corrected starting point

The research began with an apparent 8B OEF collapse to 0.270. That result compared four single-threaded workers with four copies of a four-thread isolated baseline on a four-core machine. The denominator therefore assumed four times the compute allocation actually available to each worker. Matched, pinned reruns did not reproduce the collapse (Table 1). This correction changed the research question from "can concurrency rescue an 8B bandwidth wall?" to "where should coordination and memory lifetime live?"

Table 1: Corrected four-worker baseline. Confidence intervals are those recorded by the matched-control harness.

Model	Aggregate t/s	Capacity efficiency (95% CI)	Contention efficiency (95% CI)
Llama 3.2 3B	15.479	1.099 (1.096–1.104)	0.967 (0.965–0.969)
Llama 3.1 8B	6.944	1.074 (1.069–1.079)	0.963 (0.960–0.965)
OLMoE 1B/7B	40.763	1.155 (1.141–1.172)	0.964 (0.961–0.966)

2.3 Evaluation protocol and evidence discipline

The evaluation advanced through falsifiable gates rather than a single end-to-end score (Figure 2). Each machine-readable run records the harness, model and source hashes when available, hardware description, command, warmup, sample count, and thresholds. Correctness promotion requires expert IDs to match, gate-weight maximum absolute error no greater than 10^{-6}, and output maximum absolute error no greater than 10^{-5}. The overhead gate requires coordination p99 to remain below 10% of selected-expert compute p99. Memory, admission, fairness, and prefetch experiments each use matched controls and recorded pass criteria in the canonical ledger.

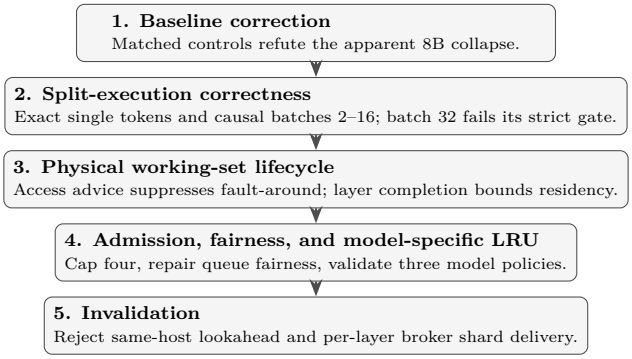

Figure 2: Evaluation arc. Each stage promoted or rejected a bounded claim before the next mechanism was introduced.

The study reports the latest provenance-complete result for each promoted claim and preserves negative results. This matters because several plausible hypotheses failed: default mmap did not track logical sparsity, token-boundary eviction did not sufficiently bound concurrent residency, FIFO did not remain fair in a deep mixed-length queue, and oracle knowledge did not make local prefetch free.

3 Correctness and Coordination Cost

Separating the graph is useful only if it preserves the reference calculation and leaves enough time for orchestration. We therefore tested numerical equivalence before promoting performance claims.

3.1 Pinned external equivalence

The reference callback captured normalized hidden states, expert IDs, routing weights, and MoE outputs from `llama.cpp` commit `6e9007ae6`. Zincrom then evaluated the corresponding split path. Three token IDs (50279, 42, and 1000) across all 16 layers produced 48 exact expert-ID matches, zero maximum gate-weight delta, and zero maximum output delta.

Causal batches 2, 4, 8, 16, and 32 were each evaluated at every layer (80 cases). Table 2 makes the boundary explicit. Batches through 16 matched exactly. Batch 32 preserved expert IDs and stayed inside the weight threshold, but layer 2 produced 1.1444×10^{-5} maximum output error, above the 1.0×10^{-5} gate. It is a recorded failure, not rounded into a pass. End-to-end autoregressive generated-token equivalence remains untested.

Table 2: External equivalence against the pinned reference across all 16 OLMoE layers.

Input	Cases	Exact IDs	Max weight Δ	Max output Δ	Verdict
3 single tokens	48	48/48	0	0	Pass
Batch 2	16	16/16	0	0	Pass
Batch 4	16	16/16	0	0	Pass
Batch 8	16	16/16	0	0	Pass
Batch 16	16	16/16	0	0	Pass
Batch 32	16	16/16	6.8545×10^{-7}	1.1444×10^{-5}	Fail

3.2 A 1,000-iteration overhead decomposition

In the matched isolated harness, selected-expert compute measured 2,205.553 μs at p99. BEAM dispatch and confirmation measured 74 μs at p99 (43.47 μs mean, 41 μs p50, 53 μs p95), yielding a 3.355% ratio. This passes the preregistered 10% gate. It supports the narrow conclusion that coordination did not dominate selected-expert computation in this path; it does not measure networked actors, distributed BEAM, weight I/O, or expert math inside a BEAM process.

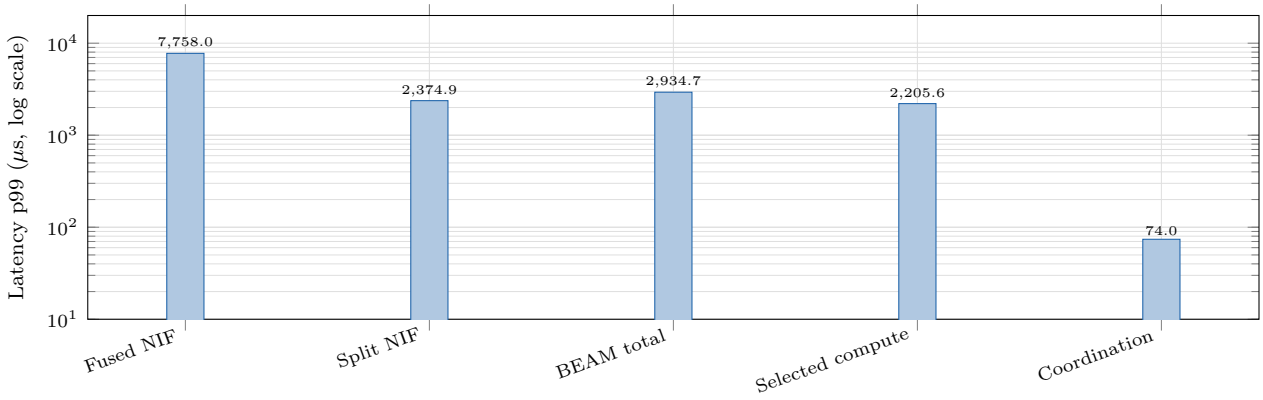

Figure 3: Latency p99 across 1,000 isolated iterations. Whole-path timings are shown beside selected compute (2,205.553 μs), the denominator for the 74 μs (3.36%) coordination result.

The same result set records fused NIF p99 at 7,758 μs, split NIF path p99 at 2,374.912 μs, and BEAM total pipeline p99 at 2,934.710 μs. These whole-path values are plotted separately from the selected-expert compute value used in the coordination ratio; conflating them would change the denominator and repeat the kind of baseline error that motivated the corrected OEF study.

Across fused, split, and BEAM paths, the recorded maximum output delta was zero. Pure isolated route timing and the no-op eight-worker dispatch result are tracked separately in the evidence ledger; neither is substituted for the full 74 μs confirmation measurement.

4 Memory Lifecycle and Physical Residency

OLMoE routes eight of 64 experts, an exact 12.5% logical expert fraction. Under file-backed `mmap`, that fraction is not a physical-memory guarantee.

4.1 The fault-around surprise

A fresh process evicted every gate/up/down expert range, verified zero present pages through `/proc/self/pagemap`, touched the routed top-eight experts at every layer, and compared them with an all-expert control. Logical active bytes were 487,587,840 of 3,900,702,720. Present bytes nevertheless reached 1,085,272,064 of 3,913,285,632 (27.7%), and RSS delta reached 1,089,163,264 of 3,921,063,936 bytes (27.8%). Normal file faults had pulled neighboring pages into the process working set.

The matched run applied `MADV_RANDOM` before the same touches. Active page presence fell to 504,152,064 bytes (12.88%), while RSS delta fell to 506,138,624 bytes against a 3,899,248,640-byte all-expert control (12.98%). The access hint removed roughly 583 MB of amplification and placed physical expert residency within 0.48 percentage points of the logical fraction. This result covers expert tensors for one routed hidden state; it excludes dense weights, KV cache, allocator peaks, and concurrency.

4.2 Why advice alone is insufficient

Eight-token decoding touches a union of experts. With one, two, and four shared-model contexts, unmanaged peak RSS deltas were 2.439, 2.881, and 3.531 GB; expert page presence rose to 60.97%, 71.32%, and 85.96%. Token-boundary eviction reduced these values, but still left 27.96%, 46.17%, and 61.43% present—above the predeclared 20%, 35%, and 55% limits—and expanded elapsed time from roughly 0.8–1.0 s to 5.2–5.9 s.

Table 3: Single-hidden-state active-expert residency. Percentages use the corresponding all-expert control.

Policy	Logical active	Present expert pages	RSS delta
Theoretical top-8/64	12.50%	—	—
Default file advice	12.50%	27.70%	1,089.2 MB (27.8%)
`MADV_RANDOM`	12.50%	12.88%	506.1 MB (12.98%)

The successful policy moves the lifetime boundary inward. A callback waits for every active request to complete MoE layer ℓ, evicts that layer only after the final arrival at the barrier, and then releases all members into layer $\ell+1$. A slower request can therefore never observe weights evicted by a faster peer. Fresh one-, two-, and four-request runs held peak expert page presence to 3.53%, 4.67%, and 6.11%, with peak RSS deltas of 223, 272, and 407 MB. Elapsed times were 1.87, 1.87, and 2.26 s: slower than accumulation, but far below the coarse token-boundary implementation.

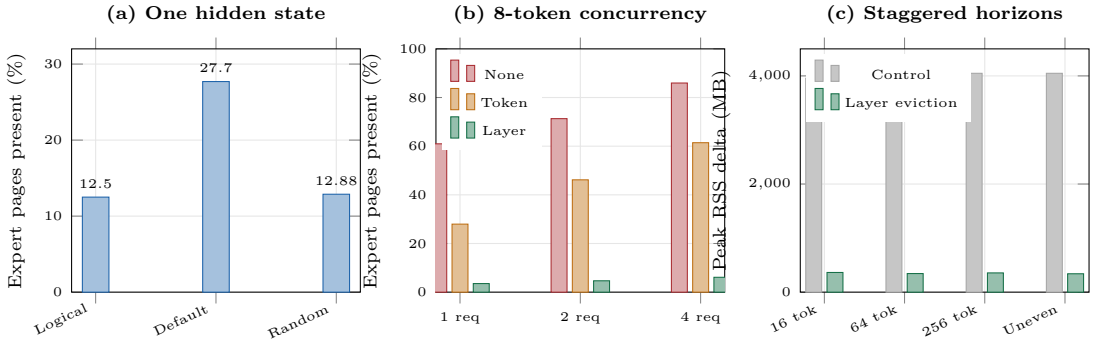

Figure 4: Physical memory behavior. (a) Default fault-around more than doubles the 12.5% logical expert fraction; `MADV_RANDOM` restores near-logical page presence. (b) Advice alone and token-boundary eviction accumulate expert pages across concurrent eight-token decodes; layer-completion eviction bounds them. (c) Layer eviction holds peak RSS nearly flat across longer and uneven staggered schedules.

4.3 Longer horizons and changing membership

The timing harness removed repeated page-table scans from the decode path and retained a 1 ms RSS monitor. Across eight staggered requests, layer eviction retained 90.8% of control throughput at 16 tokens/request, 86.4% at 64, and 86.6% at 256. The corresponding p99 latency ratios were 1.22×, 1.23×, and 1.17×; peak RSS deltas remained 364, 344, and 356 MB versus 3.957, 4.040, and 4.048 GB controls.

Uneven request lengths (64–256 tokens) changed barrier membership between token rounds. All eight requests and 1,200 latency samples completed without deadlock. Layer eviction retained 85.4% throughput, measured 1.18× p99 overhead, and held peak RSS to 340 MB versus 4.048 GB. Three seeded schedules reproduced p99 ratios of 1.176–1.184× and throughput retention of 84.6–85.3%. The mechanism is therefore supported through a 256-token retained-KV horizon and token-boundary membership changes on this host, not for arbitrary arrivals or mid-layer cancellation.

The memory result is deliberately not described as free. It exchanges page churn and synchronization for a roughly 9–15% throughput loss in the longer schedules, and its validity is tied to the measured model, kernel behavior, and four-core contention regime.

5 Admission, Fairness, and Model-Specific Policy

Bounded expert residency does not select a safe concurrency level or a fair request order. On this four-core host, eight active decodes raised throughput by 21.1% relative to four, but increased layer p99 latency by 73.1%, above the 50% admission limit. The evaluated policy therefore caps active decodes at four.

For eight simultaneous 128-token requests, FIFO cap-four reduced service p99 by 35.7% and peak RSS by 34.8%, retained 81.5% throughput, and increased end-to-end request p99 by 22.7%. The cap controlled resources, but a

deeper 16-request, 2,104-token mixed-length queue exposed starvation: FIFO's slowdown Jain fairness was 0.739, below the 0.80 gate. Aging size-aware priority—minimizing declared token budget minus rounds waited—raised fairness to 0.901, reduced wait p99 from 71.584 to 65.016 s, and raised throughput from 18.64 to 19.26 token/s. Peak RSS changed from 343 to 351 MB and remained inside the original limit. The harness supplies declared budgets; online length-estimation error remains unvalidated.

Table 4: Deep mixed-length queue: identical 16-request, 2,104-token schedule.

Policy	Jain fairness	Wait p99 (s)	Request p99 (s)	t/s	Peak RSS
FIFO cap-four	0.739	71.584	100.095	18.64	343 MB
Aging size-aware	0.901	65.016	101.746	19.26	351 MB

5.1 A cache policy cannot be inferred from top-K

An eight-token quantum removed the declared-length dependency and enabled routed-expert LRU tests. OLMoE (64 total, eight routed) favored LRU-8. Granite (40 total, eight routed) favored LRU-4. Qwen (64 routed, four selected plus four shared experts) also favored LRU-4. Granite and OLMoE select the same number of experts but favor different capacities, so the implementation uses an exact model/build allowlist and falls back to full eviction for unknown tuples.

Native route-hit counters showed 43.1% reuse for OLMoE LRU-8, 27.2% for Granite LRU-4, and 17.1% for Qwen LRU-4. The throughput gains ran in the opposite order: approximately 1.8%, 2.6%, and 4.2%. A seven-trial cold-fault probe measured median fault costs of 41.4, 153.7, and 501.6 μs. Multiplying hit rate by the median produced 17.8, 41.9, and 85.7 μs avoided per route, which matches the directional gain ordering. This three-model proxy is explanatory, not causal: OLMoE's fault distribution was bimodal, and shared-expert traffic is excluded.

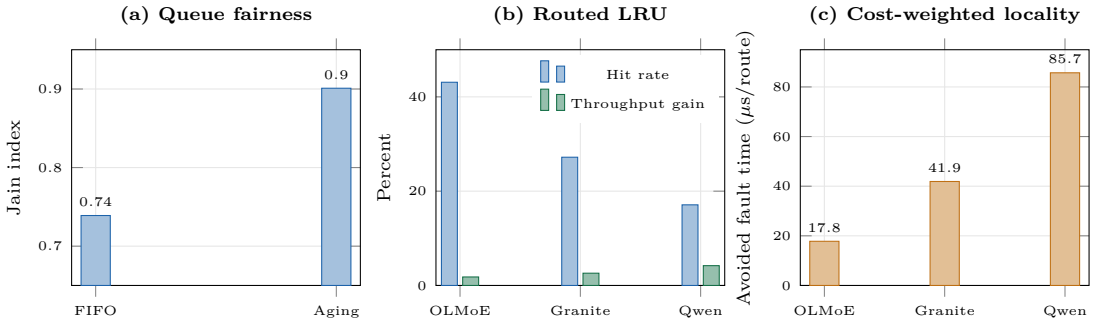

Figure 5: Policy results. (a) Aging size-aware admission repairs the FIFO fairness failure. (b) Cache hit rate and throughput gain have opposite cross-model orderings. (c) Hit rate weighted by median cold-fault time matches the gain ordering directionally; three models do not establish causality.

5.2 Context suspension is bounded, not universal

Qwen peak-memory attribution showed that parked decoder contexts, not only routed weights, were material: one configured LRU-4 run decomposed peak growth into 642.96 MB model-backed and 874.10 MB anonymous memory. F16 sequence snapshots restored logits exactly at 16, 64, 128, and 224 tokens; Q8 was 46.9% smaller but changed the top token at two of four checkpoints and produced mean absolute logit error of 0.099–0.130.

Always suspending at quantum boundaries completed all 2,104 tokens with 0.999 fairness and reduced peak RSS from 1.516 GB to 965.1 MB, but throughput was 7.189 token/s versus 7.901 for ordinary LRU-4 and 7.609 for full eviction. Larger quanta reduced reconstruction frequency. Q24 and Q28 passed their bounded nominal suites, yet reached 1.279/1.241 GB on maximum-prefix stress and 1.059/1.056 GB on 24-request bursts. Fixed-quantum suspension is therefore not a workload-independent sub-1-GB policy.

6 Negative Results: Prediction Was Not the Bottleneck

The most instructive experiments supplied the answer in advance. If oracle knowledge of the next experts could not hide page readiness, a learned predictor could not rescue the same transport path.

6.1 Perfect-knowledge local lookahead

The page-touch harness evaluated batches 1, 2, 4, and 8 across three layer transitions. Before each paired trial it evicted the selected future ranges, verified residency state, and compared concurrent lookahead with a sequential compute-then-touch control. Expert residency and numerical output gates passed, but overlap p99 was worse in seven of 12 conditions. The experiment therefore rejected the universal overlap hypothesis on the tested host.

Replacing explicit touch with `MADV_WILLNEED` failed 11 of 12 conditions. Reducing `ggml` from four compute threads to three, leaving nominal capacity for prefetch, still failed seven of 12. Finally, the compute process was pinned to cores 0–2 and a persistent mmap helper to core 3. Two nominally identical runs passed seven and 11 of 12 conditions. A preregistered three-repeat batch-eight gate then failed because one repetition regressed at layer 14→15 (15.533 ms overlap p99 versus 14.733 ms sequential).

These trials do not show that route prediction is inaccurate; prediction was perfect. They show that moving the required pages on this four-core machine competes with current compute through shared CPU, cache, and memory resources. The evidence rejects the tested same-host mechanisms and conditions, not all prefetch on larger or differently partitioned hardware.

6.2 Real shards missed the compute window

The network experiment used the three extracted 31.1 MiB selected-expert shards for OLMoE layers 0, 7, and 15. Across nine batch/layer conditions, matched selected-expert compute p99 ranged from 7.2 to 43.4 ms. Native Redpanda fetch p99 ranged from 86.96 to 119.47 ms; the persistent brod client ranged from 95.14 to 138.68 ms. No condition passed the p99 window or every-trial overlap gate. A measured 64 MiB control reached 260.95–269.44 ms p99, replacing an earlier unsupported roughly 50 ms extrapolation.

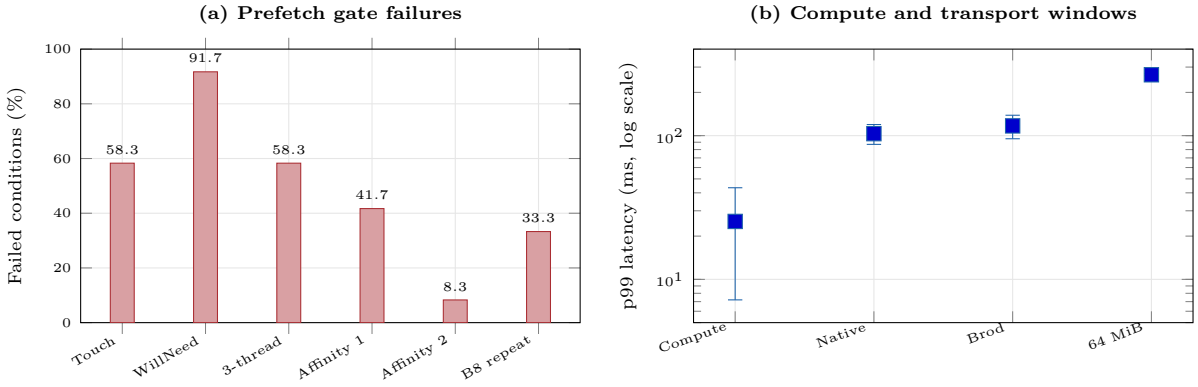

Figure 6: Failed overlap gates. (a) Fraction of failed conditions for oracle page touch, kernel advice, reserved capacity, two strict-affinity runs, and the three-repeat batch-eight gate. Denominators are 12 conditions except batch-eight repeatability (three runs). (b) Observed p99 ranges: selected compute (7.2–43.4 ms), native fetch (86.96–119.47 ms), brod fetch (95.14–138.68 ms), and 64 MiB control (260.95–269.44 ms). Error bars show ranges, not confidence intervals.

This rejects on-demand, per-layer Redpanda delivery at this shard size and hardware envelope. It does not reject networked storage in general, pre-positioning at coarser granularity, compression, faster fabrics, or hosts with independent I/O and compute capacity. The shard binaries remain private evidence and are not redistributed.

The corrected OEF result belongs to the same methodological category. A dramatic 0.270 collapse disappeared when the denominator was matched. Together, the failures argue for a research workflow that publishes gates and invalidations rather than preserving an attractive initial story.

7 Discussion, Threats, and Explicit Non-Claims

7.1 What the architecture buys

The experiments separate three concerns that are often conflated. First, orchestration can be inexpensive relative to expert compute: the measured 74 μs p99 is small in the isolated path. Second, active parameter count is not a resident-memory result: the kernel's access policy and the lifetime of routed pages determine the physical working set. Third, neither concurrency nor cache capacity is universal: admission is constrained by available cores, queue discipline affects slowdown fairness, and the value of an LRU hit depends on the avoided fault cost.

The practical architecture is therefore a sequence of explicit controls: suppress inappropriate fault-around, retire pages at the narrowest safe shared lifetime, cap active work at the measured latency boundary, age queued work to protect long requests, and select cache policy from validated model/build tuples. Unknown configurations fall back conservatively.

7.2 Threats to validity

1. **One primary host.** Core results come from a single four-core Neoverse-N1 OCI A1 instance with one NUMA node and no swap. Absolute latency, page-cache behavior, and contention may differ on other kernels, storage stacks, or CPU topologies.
2. **Model and quantization scope.** Correctness and lifecycle claims center on Q4_K_M OLMoE. Granite and Qwen extend only the routed-cache policy comparison. No accelerator result is implied.
3. **Forward-path equivalence.** Single tokens and causal batches through 16 match at all layers; batch 32 fails the strict output threshold. End-to-end generated-token equivalence remains open except for the separately scoped Q28 F16 state-transfer probe.
4. **Instrumentation.** Repeated `pagemap` scans inflated early elapsed measurements, which is why timing successors removed them from the hot path. `MADV_DONTNEED` and page-touch tests do not guarantee durable-media I/O because pages may remain in the host page cache.
5. **Statistical boundary.** Sample sizes, deterministic schedules, seeds, and pass gates are those recorded in the repository. The report does not convert host-local repeatability into population-level confidence.
6. **Transport boundary.** The Redpanda result covers real 31.1 MiB shards and the tested clients, broker, and host. It is not a claim about every network, storage service, compression format, or prefetch horizon.

7.3 Work explicitly outside this report

FFT semantic routing is inconclusive: the initial test crossed its kill threshold, and FFT magnitude over an arbitrary embedding-coordinate order discards phase. The cognitive-core proposal is conceptual and untested. Runtime 2.0 was developed after the empirical cutoff and remains active work in progress; neither its code nor its performance is evaluated here. These topics are not presented as contributions.

7.4 Provenance and licensing boundary

No model weights or extracted expert shards are included. The original upstream GGUF files are no longer present on the evaluation host, so exact upstream revisions, license records, and source-file SHA-256 values in `docs/publication/weight_manifest.json` remain unresolved. Recorded run manifests identify the files used by particular executions, but that does not establish upstream provenance or redistribution rights. A human license and notice review remains mandatory before publication.

8 Conclusion and Reproducibility

Zincrom demonstrates a bounded result: BEAM can coordinate a split sparse-MoE path without dominating selected-expert computation, but making logical sparsity real requires explicit control over page access, lifetime, admission, and scheduling. The successful mechanisms are modest and measurable—`MADV_RANDOM`, layer-completion eviction, a four-active cap, aging size-aware admission, and model-specific cache selection. The failed mechanisms are equally important: oracle local lookahead still contends with compute, and 31.1 MiB Redpanda shards arrive too late for the tested per-layer window.

The strongest conclusion is methodological. The initial OEF story was wrong; correcting it opened a more useful line of inquiry. Each successor experiment preserved a control, declared a gate, and narrowed the claim.

On a constrained machine, that discipline produced an architecture built from verified boundaries rather than assumptions about sparsity.

8.1 Reproducibility package

The local publication package maps claims to repository evidence through `docs/publication/CLAIM_EVIDENCE_MATRIX.md`. Canonical summaries and manifests for the headline correctness and overhead results are under:

- `results/sprint3-overhead-20260724T012420Z/`;
- `results/external-equivalence-20260725T031655Z/`;
- `results/multitoken-equivalence-20260725T031743Z/`.

The broader ledger is `docs/CURRENT_EVIDENCE.md`; relevant harnesses are selected into the sanitized snapshot from `zincrom_nif/` and `scripts/`. To compile this report from the repository root:

```
cd docs/publication/paper
SOURCE_DATE_EPOCH=1784937600 tectonic zincrom_technical_report.tex
```

To rebuild and verify the sanitized review snapshot:

```
SOURCE_DATE_EPOCH=1784937600 \
  bash docs/publication/build_sanitized_snapshot.sh
cd docs/publication/sanitized_snapshot
sha256sum -c SNAPSHOT_MANIFEST.sha256
```

The package enables evidence inspection and source-level reconstruction of the harnesses; it does not by itself reproduce the historical OCI environment. Base weights must be acquired separately under their upstream terms, and exact upstream revisions, licenses, and source hashes remain pending human verification. The technical report and its version history are available through Zenodo concept DOI 10.5281/zenodo.22741249. No GitHub release tag, Hugging Face publication, or public website deployment is finalized.

Repository Artifact References

Each empirical reference below is a path in the Zincrom repository; no external measurement is used to support the results.

E1 `docs/CURRENT_EVIDENCE.md`: canonical experiment ledger and promoted/rejected claims.
E2 `docs/publication/CLAIM_EVIDENCE_MATRIX.md`: claim-to-artifact traceability record.
E3 `docs/ZINCROM_CLAIMS_TEST_RESULTS_SUMMARY.md`: phase-close test index.
E4 `results/sprint3-overhead-20260724T012420Z/`: 1,000-iteration split-path overhead result.
E5 `results/external-equivalence-20260725T031655Z/`: 48-case single-token equivalence result.
E6 `results/multitoken-equivalence-20260725T031743Z/`: 80-case causal-batch equivalence result.
E7 `results/track2_test1_verdict.md`: router and top-K split feasibility verdict.
E8 `zincrom_nif/`: NIF sources and experiment harnesses used by the evaluated system.