Beyond Transformers: Sums, Maxima and the Price of Order

Neil D. B. Bruce *brucen@uoguelph.ca*
University of Guelph

Abstract

The properties that make a transformer work were read from the transformer, but they describe a design space, not a design. Any machine in which content decides who talks to whom, something else decides what is said, and the step repeats faces seven conditions, each stated within a declared class of machines: a tether to its own input (the anchor), a routing budget, a null (abstention), separate address and cargo (alignment), gain below its tether (gain control), an arrow and a writer. We price six of the conditions with results proved within declared classes and show for the seventh that transport alone cannot create features, and two of the conditions organize the space. Against an exchangeable background, a decision that sums over all N items needs a $\ln N$ schedule, while one that reads a maximum needs none against backgrounds bounded below its threshold and stops only at an evidence wall that the keys set and that, under a condition on the joint likelihood of the scores, no router can move. A hard machine orders writes without spending any of the router's margin, while a soft router with $K \ln r$ nats of positional range orders at most a fraction of pairs that tends to $K/(K+1)$, where two identical writes count as ordered when the later gets at least r times the earlier's weight. In mean field, if every position must stay retrievable, the positional terms are paid from the router's margin Δ above the gain wall: their exponential mean over positions is at most e^{Δ}, so at the gain wall nothing can be ordered. Beneath both, routing information is conserved in the sense of a family of lower bounds: naming one of N items costs about $\ln N$ nats in any machine, and only the currency differs. Registered tests on constructed keys and codes measure the walls and the exchange rates, and a catalog places 43 families, from attention variants to physical and social analogies, on the conditions. From the law of sums we build a max-statistic head that should abstain and retrieve at every length with no schedule, up to its evidence wall. In trained 125M language models with positions interpolated by YaRN, it is the only rule tested whose null and match weights both stay within half a nat of log-odds out to eight times the training length, on seed means though not in every seed, and it fails without interpolation: the matching key falls far down the ranking, and the background maximum grows with length. These are attention weights, not behaviour. Seven small machine families train end to end as the token mixer of otherwise identical small models, but a log-N scale learned in training does not remove the length penalty of the machines that normalize competition and were trained with it. We read the transformer as the cheapest point under one price list, not the only point.

1 Introduction: the lattice is a design space

A transformer block does several jobs at once. Its attention decides which tokens talk to which, its value and output maps decide what is said, its residual path keeps each token tied to its own input, and the step repeats down the stack. The protection lattice of Bruce (2026d) lists six properties that such a block must keep, each a conditional statement within a declared class of models: what goes wrong, and at what price, when a model lacks it. It also reads softmax attention in a pre-normalized residual block as the point where all six are cheap together under one price list, dense matrix multiplies and gradient descent. That reading is a judgment about prices, and the protections themselves need no softmax, matrices or layers. The monograph (Bruce, 2026a) is the collection's reference text: it develops the operator theory beneath the protections and holds the companion results cited here.

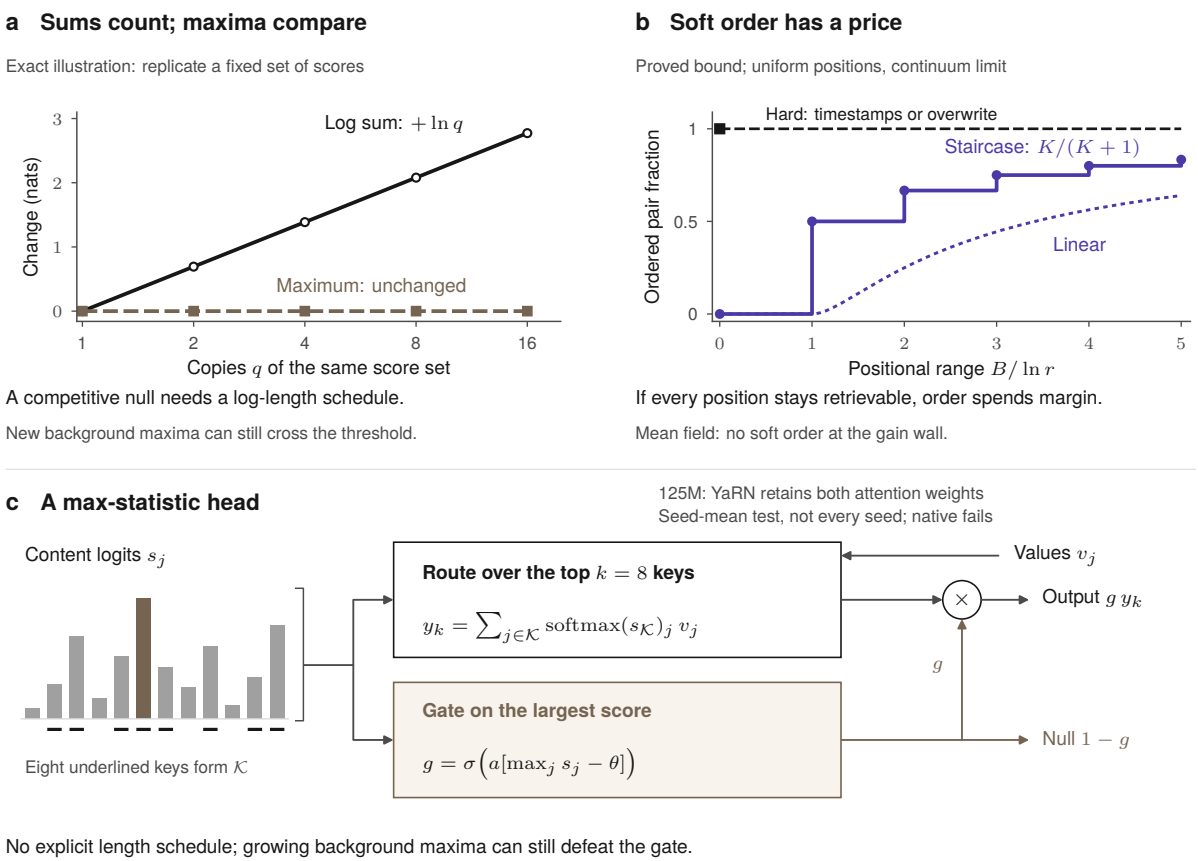


Figure 1: **Sums pay for population size; a max-statistic head separates routing from abstention, while soft order has a positional price.** Exact illustrative constructions, not experimental measurements. **(a)** Replicating a fixed score set q times adds $\ln q$ to $\ln Z$, $Z = \sum_j e^{s_j}$, and leaves $\max_j s_j$ unchanged. A competitive null's logit must grow by $\ln q$ to keep its mass fixed. New extreme background scores can still cross a maximum-based test's threshold. **(b)** For two identical writes, ordered means that the later has at least $r > 1$ times the earlier weight. With positional range B, $K = \lfloor B/\ln r \rfloor$, the optimal equal-block staircase orders $K/(K+1)$ of pairs in the uniform-position continuum limit; a linear bias orders $(1 - \ln r/B)_+^2$. Hard timestamps or overwrite order every pair without spending positional-logit margin. The gain-wall statement additionally assumes mean field and retrieval of a single target at every position (Proposition 11). **(c)** Illustrative logits feed the specified top-k route ($k = 8$, underlined keys) and a largest-score gate. Values v_j enter separately; σ is logistic and a, θ are learned per head. The output has mass g and the null has mass $1 - g$. No explicit length schedule is built in. In trained 125M language models, the registered secondary retention test for null and match weights passed with YaRN on mean per-seed log-odds changes over three seeds, but not in every seed. Native extrapolation failed as the matching key fell far down the ranking and the background maximum grew with length. These are attention weights from fallback heads, not behavioural abstention or retrieval, and give no guarantee beyond the tested lengths (Figure 10 and Section 11).

This paper takes the protections at their word and asks what else they allow. Stated without the transformer's vocabulary, they become conditions on any machine in which content decides who talks to whom, something else decides what is said, and the step repeats. Seven conditions result (Section 2): a tether to each token's own input, a routing budget, a way to route nowhere, separate channels for address and cargo, loop gain below the tether, an arrow that tells later writes from earlier ones, and a writer, because transport alone cannot create features (Proposition 10). Each condition is a test. Removing it leaves a signature that a measurement can see, and a machine can pay for it in more than one currency. A soft router pays the routing budget in logit gain, a walker in samples, a tree in depth and an address book in bits. We price these payments with propositions, each proved within a declared class of machines (Appendix A), and test them on constructed keys.

Two results organize the space. The first is a law of sums. A decision that sums over all N items, such as a softmax normalizer, a sink that competes with it or a sampler's competitive killing, needs a $\ln N$ schedule against an exchangeable background: its logit scale or its null must grow like $\ln N$ to keep the same decision as N grows (Propositions 2 and 3). A decision that reads a maximum, such as a key check, a gate on the largest score or a hard stage, needs no schedule against backgrounds bounded below its threshold (Figure 1a). It stops only at the evidence wall, where the background maximum reaches the target's score or, for a test that abstains, its threshold (Section 4). That wall belongs to the keys. Under the premises of Proposition 1, no router, soft, sampled or hierarchical, finds the target more often than the argmax does, so samples, trees and gain can bring a router up to the wall but never past it. For random keys the wall grows with the dimension at the capacity of a Gaussian channel, and a registered test with explicit keys finds that rate, not its small-signal approximation (Section 3).

The second is a price of order. A machine that reads a store written over time often needs the latest write, not just a matching one. A hard machine orders writes without spending any of the router's margin, by comparing timestamps or by overwriting. A soft router can order two identical writes only through a positional term in its logits. With $K \ln r$ nats of positional range it orders at most a fraction of pairs that tends to $K/(K+1)$. In mean field, if every position must stay retrievable, the positional terms are paid from its margin above the gain wall, their exponential mean over positions being at most the exponential of that margin (Proposition 11, Figure 1b), so at the gain wall it can order nothing.

Beneath both laws sits a conservation of routing information, by which we mean a family of lower bounds (Proposition 14). Naming one of N items costs about $\ln N$ nats whatever the machine, and machines differ only in the currency they pay in: key width, logit range, state or positional range. Each bound holds within its own class of machines and prices one currency there, and the bounds set no exchange rate between currencies. Together they say that no fixed allotment of any one resource survives a growing N (Corollary 15). Attention pays for memory in full and buys recency from its positional range; a fixed-state recurrence gets direction and recency by construction and must pay for memory (Table 2). Three registered tests measure how efficiently machines pay (Section 3).

The conditions do more than describe the transformer: the law of sums says how to build a head it lacks. A max-statistic head routes by a softmax over its top k keys and abstains through a gate that compares its largest score with a learned threshold, so neither of its decisions sums over N (Figure 1c). It should keep abstention and retrieval at every length with no schedule, up to its evidence wall. Trained in 125M-parameter language models, it was the only rule tested whose null and match weights both changed by less than half a nat of log-odds from the training length to eight times it when positions were interpolated with YaRN, on seed means though not in every seed, while five rules kept the match weight alone better; and it failed without interpolation as the matching key fell far down the ranking and the background maximum grew with length (Section 11). These are attention weights, not behaviour. The price of order adds a smaller design rule: a soft recency term should be a staircase, and exact order should come from a hard mechanism.

The rest of the paper maps the space. A catalog places 43 families, from attention variants to physical and social analogies, on the seven conditions, each mapping checked against its sources (Section 9). Seven small machines train end to end as the token mixer of otherwise identical small models, but a log-N scale learned in training does not remove the length penalty of the machines that normalize competition and were trained with it, so their length transfer is reported machine by machine (Section 10). A readout battery finds no

head in those small language models that meets its threshold for a retrieval head, and finds the trained heads far outside the region that the gain certificate covers, read with their measured update shares, which, the certificate being sufficient only, does not make them multistable (Section 11). Throughout, the conditions price protections; they do not predict capability. Which point of the space is cheapest depends on prices that this paper does not measure on a common budget. On the lattice's reading, the transformer is the cheapest point under the dense-matmul price list, not the only point (Section 12).

Contributions.

- Seven conditions that do not assume a transformer, each stated as a test with its removal signature and its currencies, six of them priced by results proved within declared classes and the seventh shown, since transport alone cannot create features (Section 2, Appendix A).
- The two limits of routing, which Bruce (2026c) state for softmax (the dispersion of a bounded softmax is known: Veličković et al., 2025), extended to any router: an evidence wall set by the keys, which for random keys grows with dimension at the capacity of a Gaussian channel, and a gain wall set by the read. The law of sums separates the decisions that pay the second from those that do not. Registered tests measure both limits with explicit random keys and in trained lookup heads (Sections 3 and 4).
- The price of order, proved: a staircase of positional terms attains the bound, a linear bias leaves about twice as many pairs unordered, and in mean field the positional terms are paid from the margin above the gain wall (Section 8).
- The conservation of routing information, a family of lower bounds: five ledgers, a wall for constant resources, and three registered tests of the exchange rates (Section 3, Appendix A.10).
- Tethers priced for hard and soft routers, gain control for softmax routing and separation for both: coalescence that spread does not see, a certificate for uniqueness, and the exact cost of an address that carries cargo (Sections 5 to 7).
- A max-statistic head built from the law of sums and tested in trained 125M language models; small trained tests of seven machine families; a readout battery for trained walls; and a catalog of 43 machine families (Sections 9 to 11).

2 Seven conditions that do not assume a transformer

The protection lattice of Bruce (2026d) lists six protections, each a conditional statement within a declared class of models, on the operator theory developed in Bruce (2026a, Chapters 1–4): what goes wrong, and at what price, when a model in that class lacks the property. None of them needs softmax, matrices or layers. Stated without those words, they become conditions, each stated within a declared class, on any machine in which content decides who talks to whom, something else decides what is said, and the step repeats: a machine that lacks one shows the matching failure on tasks that need that protection. A seventh condition joins them: transport alone cannot create features, so every machine needs a writer.

Each condition is a test. Removing it leaves a signature that a measurement can see, and a machine can pay for it in more than one currency. Table 1 lists the seven, with the proposition that prices each. Every proposition is proved within a declared class: it is stated with its premises, and its proof is in Appendix A. The measurements in Sections 3 to 8 are constructed tests that check the propositions, not trained-model results, except where a section says otherwise. Two results organize the rest. The law of sums (Propositions 2 and 3) says that a decision summing over all N items at a fixed logit scale needs a $\ln N$ schedule against exchangeable backgrounds, while a decision that reads a maximum needs none against backgrounds bounded below its threshold. In the Gaussian surrogate of Section 3, a sum whose scale grows past the point where the background sum is carried by its largest terms reads the maximum, at the price of a scale that grows with N. The price of order (Proposition 11) says that a hard machine orders writes without spending any of the router's margin, while a soft one pays in positional range.

Condition	Removal signature	What to measure	Currencies	Priced by
A tether (anchoring)	Soft routing reaches consensus; hard routing coalesces onto its cycles	Identity retention; spread; distinct states	Restart probability, residual share, leak conductance, retained endowment	Proposition 4
A routing budget	Dilution (soft) or misses (hard) once $\ln N$ outgrows it	The failure onset $\ln N^*$ against the currency	Logit gain; $\ln M$ samples with a select readout; tree depth; address bits; and the query's evidence, which no router replaces	Propositions 1, 2 and 14
A null (abstention)	Forced answers: absent queries retrieve something	False retrieval on absent queries against N	A competitive null logit; an absolute threshold; an empty bucket; an outside option	Proposition 3
Separate address and cargo (alignment)	Contamination: routing scores pick up payload terms	The onset shift, tied against untied	Channel width; factorized addresses	Theorem 6
Gain below the tether (gain control)	Runaway: several fixed points, high sensitivity, lock-in	Contraction, multiplicity, sensitivity	Logit scale times squared radius, relative to the anchor; comparator sharpness	Proposition 5
An arrow (direction; for a store written over time, write order)	Order blindness: soft routers fail every latest-write query, hard ones half	Latest-write accuracy, and its price on old items	Positional range; masks; timestamps; overwrite; driven cycles	Proposition 11
A writer	No new features: nonlinear readouts such as XOR fail	Accuracy on feature-creation tasks	MLPs, edge messages, lookup tables, nonlinear devices, reactions	Proposition 10

Table 1: The seven conditions as tests. The first six are the protections of Bruce (2026d), named there as in parentheses; the writer is the seventh. Its removal signature is a prediction that this paper does not test: Proposition 10 keeps transport's outputs in the convex hull of its input rows, at no higher rank, but gives no accuracy bound without a specified readout. A machine meets a condition by paying in any of its currencies, and the result named in the last column prices the payment. Sections 3 to 8 develop the first six.

3 The routing budget is two budgets: evidence and gain

A router must name one of N items from the scores it computes against them. Two different things can stop it. The keys may not carry enough evidence: some background item may score as high as the target. Or the read may not express what the keys carry: a soft router must put its mass on the target against N competitors. The first limit belongs to the keys and the second to the read. In the language of decoding they are the code limit and the decoder limit (Shannon, 1959; Forney, 1968). Every machine's routing budget has both parts, and only the second depends on the machine.

Evidence. Write $s_1, \dots, s_N$ for the scores and t for the target.

Proposition 1 (Evidence cannot be bought)**.**

(a) If a soft router with weights proportional to $e^{\tau s_j}$, $\tau > 0$, puts at least half its mass on the target, even with an extra nonnegative null weight in its normalizer, then the target is the argmax: $s_t \geq \max_{j \neq t} s_j$.

(b) Let the target be uniform over the N items, and let the joint density of the scores, given that the target is t, factor as $p_t(s) = g(s)\,\ell(s_t)$ with ℓ nondecreasing. Independent background scores with density f_0 and an independent target score with density f_1, with f_1/f_0 nondecreasing, are an example. Then the argmax is the MAP rule: no rule that sees only the scores, randomized or not, finds the target more often.

(c) If the $N-1$ background scores are independent with continuous distribution function F and the target scores c, the target is the argmax with probability $F(c)^{N-1}$. If $0 < F(c) < 1$, its real-valued half-success crossing is $N_{\mathrm{ev}} = 1 + \ln 2/(-\ln F(c))$.

Part (a) needs nothing. Part (b) needs its joint condition: exchangeable backgrounds with a monotone marginal likelihood ratio are not enough, and Appendix A.1 gives a three-item counterexample. Part (c)

is the argmax's exact curve for a fixed target score, not a ceiling by itself, since a rule that knew c exactly could pick the item that scores c. For a random target score, averaging (c) over it gives the argmax curve, and under (b) no rule beats that curve. So, under (b), samples, trees, schedules and more gain can bring a router up to the argmax curve but never past it, and only better keys move it.

For random keys the onset has a closed form. Let the background keys be uniform on the unit sphere in $\mathbb{R}^D$ and let the target's key have cosine $c \in (0,1)$ with the query. Then $N_{\text{ev}} \approx \ln 2/(1-F_D(c))$, and by Proposition 14(b)

$$-\ln\big(1-F_D(c)\big) = (D-1)\,C(c) + \ln\big(c\sqrt{2\pi(D-1)}\big) + O(1/D), \qquad C(c) = -\tfrac{1}{2}\ln(1-c^2) = \tfrac{1}{2}\ln(1+\text{SNR}),$$

with $\text{SNR} = c^2/(1-c^2)$. The onset grows with the dimension at the capacity of a Gaussian channel at that signal-to-noise ratio (Shannon, 1959), not at its small-c form $c^2/2$. Random keys form a spherical code, and the evidence wall is the limit of decoding it.

Measuring the code limit. A registered test drew explicit random keys: 400 trials in each of 20 settings of c and D, with the registration frozen before the run. In every setting the argmax's measured onset matched the exact law $F_D(c)^{N-1}$, with largest $|z|$ 2.80 against a family-wise threshold of 4.06. Its slope in D, after the prefactor $\ln(c\sqrt{2\pi(D-1)})$ is removed, was the capacity, not its leading order: 0.047, 0.107, 0.211 and 0.499 at $c = 0.30$, 0.44, 0.60 and 0.80, against $C(c) = 0.047$, 0.108, 0.223 and 0.511, where $c^2/2$ would give 0.045, 0.097, 0.180 and 0.320. The two rates are close at small c; the registration names $c = 0.6$ and 0.8 as the values that tell them apart. The wall is close at practical sizes. At $D = 128$ the strongest of a million random keys has median cosine 0.408 over 100 trials (exact median 0.410). Among a million keys, a target matched more weakly than that is outscored at least half the time by the best background key, and then no reader that ranks the scores can find it.

Gain. Soft routers spend a second budget. Fix the scores, with no ties at the target's.

Proposition 2 (Gain can be exchanged)**.**

(a) Select. *Draw M independent samples from a routing distribution that gives the target mass w and the items scoring above it total mass q, and keep the sample with the largest score. This finds the target with probability $(1-q)^M - (1-q-w)^M$. When $q = 0$ it succeeds at least half the time exactly when $w \geq w_M = 1 - 2^{-1/M}$.*

(b) Average. *A readout that needs a strict majority of the M samples on the target succeeds with probability $\Pr\{\text{Bin}(M,w) > M/2\}$. Its half-success threshold in w is never below $\frac{1}{2}$: exactly $\frac{1}{2}$ for odd M and higher for even M (0.614 at $M = 4$).*

(c) Trees. *If every internal node weights its children by their subtrees' partition functions, the leaf probabilities equal the flat softmax for any scores. A two-level product shortlist with $k \geq 2$ entries per half that contains the target in both halves, with target half-scores $c/2$ and other shortlisted half-scores at most u, puts at least half the mass on the target over this bounded-score class exactly when $\tau(c-2u) \geq 2\ln\big((1+\sqrt{2})(k-1)\big)$.*

In the mean-field model, N counts background items, and their sum is replaced by $Nm(\tau)$, with $m(\tau) = \mathbb{E}e^{\tau z}$. There the soft router's majority onset is $\ln N^* = \tau c - \ln m(\tau)$, its gain wall. Selection moves the wall to $\tau c - \ln m(\tau) + \ln(1/w_M - 1)$. Since $w_M \approx \ln 2/M$, each nat of $\ln M$ buys a nat of $\ln N$. Averaging buys nothing: by (b) each query still needs a majority of its mass on the target. By (c), a tree whose nodes weight their children by subtree partition functions is the flat softmax, so a tree's advantage lives in its hard stage. Even so, a soft final stage over a $k \times k$ shortlist needs $2\ln((1+\sqrt{2})(k-1))$ nats of separation: 3.96 at $k = 4$ and 7.18 at $k = 16$. (Arbitrary learned parent scores do not reduce to the flat softmax, and Appendix A.2 gives an example.)

Two premises bound these statements. The threshold w_M, and with it the select wall, needs $q = 0$. When sampled distractors can outrank the target, $(1-q)^M \to 0$ and more samples eventually hurt. The mean field is certified only where the background sum concentrates. For independent background scores with $m(2\tau) < \infty$,

Chebyshev's inequality bounds the probability that its relative error exceeds u by $(m(2\tau)/m(\tau)^2 - 1)/(Nu^2)$. At large τ the sum is dominated by its few largest terms. For Gaussian scores $z \sim \mathcal{N}(0, 1/D)$ the mean-field onset $\tau c - \tau^2/(2D)$ peaks at $\tau = cD$; beyond that the background sum freezes, and the true onset keeps rising toward the evidence wall instead of falling (Lucibello & Mézard, 2024). That is an explanation within the Gaussian surrogate, not a finite-sample theorem for bounded keys. One part holds for any keys at every N: a majority reached at some τ is kept at every larger τ, so the true onset never falls as τ grows.

Constructed-key records. The package's product substrate stores keys as two 64-dimensional halves ($D = 128$) and queries them exactly ($c = 1$). A key's score is the mean of its two half-cosines, so the flat softmax factorizes over the halves, as in the proof of (c). The mean-field walls of Proposition 2 for softmax and for a select walker with M samples then become

$$\ln N^* = 2\ln\bigl(1 + (\sqrt{2} - 1)\, e^{\tau c/2 - \ln m(\tau/2, 64)}\bigr), \qquad \ln N^*_M = 2\ln\bigl(1 + (w_M^{-1/2} - 1)\, e^{\tau c/2 - \ln m(\tau/2, 64)}\bigr),$$

with $m(\cdot, 64)$ taken over one half. Softmax's wall sits there: measured at $\ln N = 7.90$, 12.36 and 16.67 at $\tau = 10$, 15 and 20, against 7.89, 12.37 and 16.69. Selecting walkers gain 1.066 and 1.070 nats of $\ln N$ per nat of $\ln M$ over $M = 16$ to 4,096 at $\tau = 5$ and 10, with every onset within 0.07 nats of $\ln N^*_M$. Averaging walkers do not move the wall outward, and with $M = 4$ they move it inward, since by (b) their half-success threshold at $M = 4$ exceeds one half. The registered prediction that averaging matches softmax at every M therefore fails (Figure 2a). A tree with a hard top-four shortlist per half shows no gain wall up to 2^{30} slots at $\tau \geq 10$. With a top-16 shortlist, its soft final stage still fails at $\ln N = 12.37$ at $\tau = 10$, consistent with (c). With weaker queries the machines' onsets approach the hard argmax's evidence wall, at $\ln N = 8.372$ for $c = 0.35$ and 18.006 for $c = 0.5$. Select walkers with enough gain and samples reach it to within the grid's resolution. The registered convergence of the high-currency machines is only partly met: at $\tau = 120$ softmax stays 0.23 and 0.81 nats short, against a registered tolerance of 0.5. These tests fix c, so they lie outside the premises of Proposition 1(b), and how close a walker comes to the wall is observed here, not implied by that proposition.

Recovered product-key measurements. Figure 2 shows the recovery's fine-grid measurements. The selection slopes are 1.066 and 1.070 at $\tau = 5, 10$; the $\tau = 15$ slope is unscored because crossings are censored. The registered primary check, that every historical verdict keeps its status on the fine grid, fails: seven historical statuses change, all in the registered comparison of averaging walkers with softmax at $\tau = 15, 20$. Every check of the select walkers keeps its status, and no new fit or assessment is made here. The recomputation and its checks are described in Appendix C.2.

The two limits. The two budgets are the two limits of decoding. The code limit is the evidence wall: it depends on the keys, through D and c, and under part (b) of Proposition 1 no read moves it. The decoder limit belongs to the read. A soft read whose logits lie in a window of width R can put a majority on one of N items only if $\ln(N-1) \leq R$ (Proposition 14(c)), the dispersion that Veličković et al. (2025) prove for bounded softmax. Raising τ, or growing it with $\ln N$, widens the window and lifts the decoder limit toward the code limit; in the registered test it never passed it. A read that takes the maximum over all items has no decoder limit. The registered test measured both. Raising τ from 2.5 to 80 moved softmax's onset from 3.4–10.0 nats below the evidence wall to 0.02–0.81 nats below it, never above. Its onset followed $\tau c - \ln m(\tau, D)$ in all 57 registered comparisons inside the mean-field regime, with largest deviation 0.40 nats against a band of $\ln 2$.

In trained heads. A second registered stage trained lookup heads on noisy-match geometries, with D varied: four rules, logit budgets $\tau_{\max} \in \{10, 20, 40\}$ and three seeds, 360 cells in all. It reads accuracy and weight separately. On accuracy, no rule outlasts the code limit by more than one grid step, and a head that reads the maximum (a key check) and a tempered sum reach the limit at every D, to within one grid step. Their slopes in D, with the prefactor removed as above, are 0.215 and 0.216 against $C = 0.223$ at match cosine 0.6, and 0.498 and 0.507 against 0.511 at 0.8, where $c^2/2$ would give 0.18 and 0.32. A sum at a fixed scale falls behind as D grows. At $\tau_{\max} = 10$ its accuracy onset sits 0.12–0.16 nats below the code limit at the smallest D and 1.06–1.44 nats below it at the largest. These gaps are seed means of crossings interpolated on a grid of 0.69-nat steps, reported but not scored; the scored test found the sum at least one grid step short at the largest D in every seed. On weight, the decoder limit binds a trained sum. Of the 220 scored cells, the 11

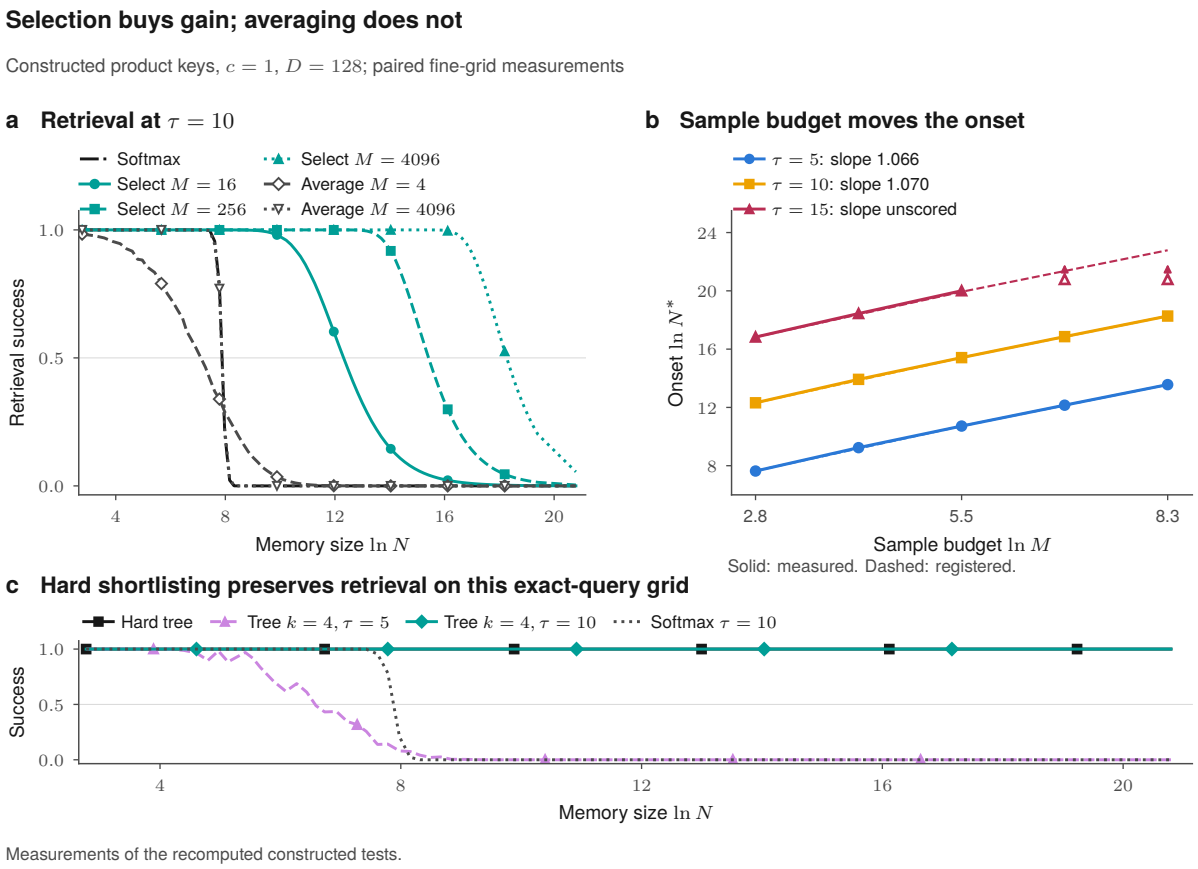


Figure 2: **Selection exchanges samples for retrieval capacity; averaging does not.** Recovered constructed product keys have $D = 128$ and exact queries ($c = 1$). (a) Retrieval success at $\tau = 10$: softmax, selection with $M = 16, 256, 4096$, and averaging with $M = 4, 4096$. (b) Saved first downward 50% crossings, interpolated in $\ln N$, versus $\ln M$ at $\tau = 5, 10, 15$; dashed lines are registered predictions. Open symbols and upward arrows mark no crossing by $N = 2^{30}$. Saved slopes are 1.066 and 1.070 at $\tau = 5, 10$; the $\tau = 15$ slope is unscored. (c) Hard tree, top-four-per-half soft trees at $\tau = 5, 10$, and softmax at $\tau = 10$. Softmax and soft-tree success require half the target mass; averaging requires a strict sample majority. Each point pools three seeds with 2,000 queries each; conditional walker probabilities are evaluated exactly. No seed intervals were retained. Source: the constructed-test recovery (Table 11). On the finer grid seven historical verdicts change, all for averaging walkers against softmax (Appendix C.2). These are synthetic tests.

in the concentration regime fall on the trained law, two-sided within a grid step. The rest fail no earlier than one grid step before it, as Jensen's inequality allows: outside that regime the law bounds the failure on one side only. On abstention, in a post hoc reading of the saved cells, tempering does not rescue these heads. Its factor is one at the training count of 64 keys, so in all 90 pairs the tempered head has its fixed-sink partner's weights, and its absent-query abstention falls with N as the fixed sink's does: from a median of 0.59 at the training count to at most 0.005 at $N = 65{,}536$, against at most 0.013 for the fixed sink. With exact queries the tempered sink keeps abstention over the tested range (Section 4); with these noisy matches it does not.

Routing information is conserved. Whatever the machine, naming one of N costs about the same nats, as a family of lower bounds, one for each class of machines; only the currency differs. Proposition 14 prices five ledgers, each within its own class of machines, and Appendix A.10 proves it.

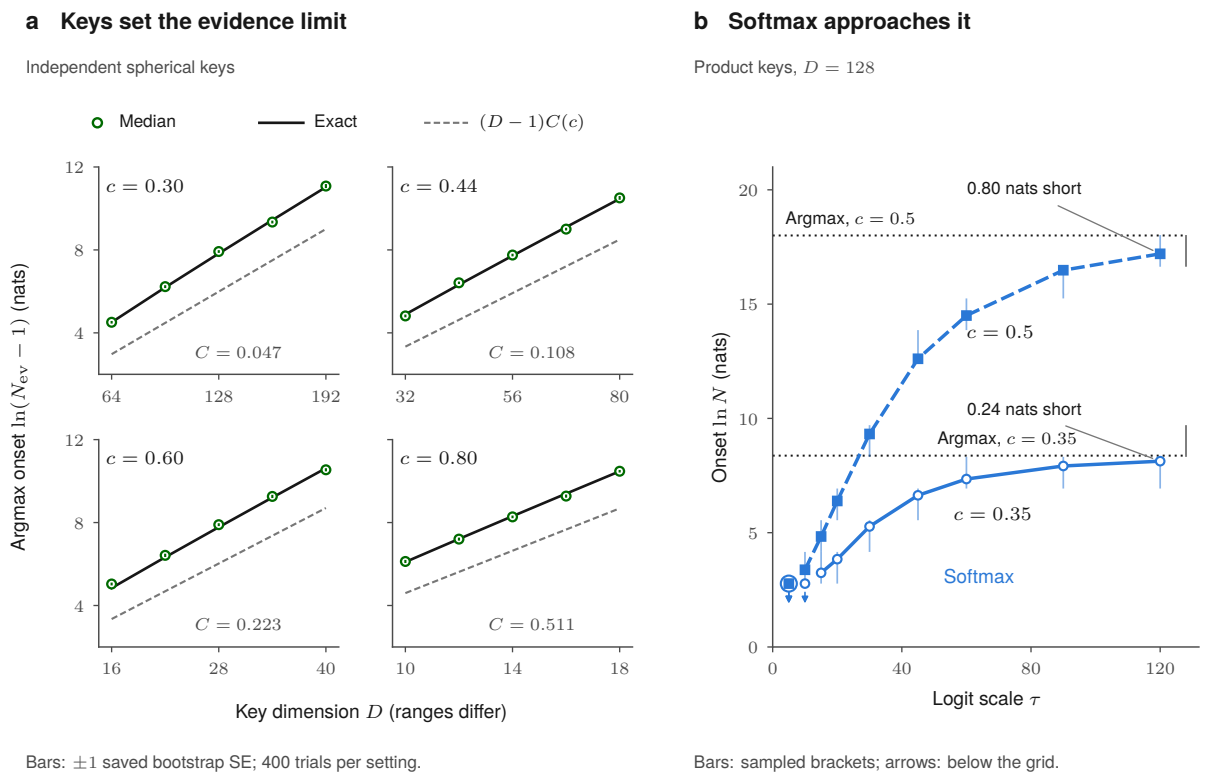


Figure 3: **Keys set the evidence limit, while increasing softmax gain approaches it on these synthetic substrates.** (a) Independent spherical keys (the evidence-capacity test): all 20 settings, with 400 trials each. Open circles show the log median first background exceedance T_{ev}; bars are ±1 saved bootstrap standard error (2,000 resamples, seed 29), often smaller than the symbol. This background-count convention is $\ln(N_{ev} - 1)$, matching the exact half-success law $N_{ev} - 1 = \ln 2/[-\ln F_D(c)]$ (solid). Dashed lines show the capacity term $(D-1)C(c)$, where $C(c) = -\frac{1}{2}\ln(1-c^2)$; finite-size prefactors separate it from the exact law. Reference lines join values at the tested dimensions; the four facets share their vertical scale but use different dimension ranges. (b) The separate product-key retrieval test ($D = 128$, two 64-dimensional halves): softmax's first downward 50% success crossing, interpolated in $\ln N$, against gain τ. Success requires at least half the target mass. Dotted lines are the measured argmax crossings. Blue bars and the rightmost grey bars span the sampled sizes bracketing softmax and argmax crossings, respectively; they are grid-resolution brackets, not confidence intervals. Downward arrows mark failure already at $N = 16$: both cosines at $\tau = 5$ (nested symbols), and $c = 0.35$ at $\tau = 10$. Each success value pools three seeds × 2,000 queries; seed-level spread was not retained. At $\tau = 120$, softmax remains 0.24 and 0.80 nats below the respective walls on this historical coarse grid (0.23 and 0.81 on the recomputed fine grid, Figure 5). Sources: the evidence-capacity test and the historical product-key records (Table 11).

- *Identification.* Naming the target with error at most ε takes at least $(1-\varepsilon)\ln N - \ln 2$ nats about it (Fano's inequality).
- *The key channel.* Readers that rank keys succeed only through the argmax, at the code limit above.
- *Expression.* A soft majority needs a logit range of at least $\ln(N-1)$.
- *Memory.* A one-pass state that answers after reading N items of v nats each, returning each value with error at most $\varepsilon \le 1 - e^{-v}$ (no worse than guessing), needs $N(v - h(\varepsilon) - \varepsilon\ln(e^v - 1))$ nats, where h is the binary entropy. Membership with no false negatives and a false-positive rate ε needs about $N\ln(1/\varepsilon)$.

Machine	How it pays	Where it stops
Softmax attention	Memory in full, with a cache that grows with N; expression with its logit scale or a $\ln N$ schedule; recency from positional range	Its decoder limit unless the scale grows; then the code limit
Fixed-state recurrence	Direction and recency by construction	The memory wall, once $N(v - h(\varepsilon) - \varepsilon \ln(e^v - 1)) > S$, unless the question comes first
Max-statistic head	A constant $\ln(k-1)$ for expression, independent of N: it selects	The code limit
Timestamps	Order, with $N \ln N \geq \ln N!$ nats of storage	—
Overwrite	Never pays for order: it answers latest-value queries only	It loses the history

Table 2: Who pays which ledger of Proposition 14. S is the state size in nats and v the nats per stored value. Every machine that ranks keys stops at the code limit at the latest.

- *Order.* A recoverable order of N writes costs $(1-\varepsilon)\ln N! - \ln 2$ nats. A soft router pays for order in positional range instead (Proposition 11).

No fixed allotment of dimension, logit range, state or positional range survives a growing N (Corollary 15), and each wall moves only when its own ledger is paid. Table 2 shows who pays which.

Measuring the exchange rates. Three registered tests measured how efficiently machines pay: a memory test, a positional-code test and a ledger-failure test, each registered before its run, with one later amendment to the memory test that only made its assessment stricter.

- *Memory.* A linear-attention (outer-product) state of d^2 numbers, written with random unit keys in $\mathbb{R}^d$ and independent values uniform on $\{-1,+1\}^d$, and read by signs, recalls 0.315–0.328 nat per number, close to $1/\pi$, once it holds $16d$ items or more. A least-squares state, the delta rule's fixed point, recalls every stored bit exactly up to $N = d$, which is d^2 bits at $N = d$, and then falls to the same rate. The online delta rule with a unit update rate peaks at 0.272–0.275 nat per number at $N = d$ to $2d$, settles at 0.264–0.267 by $N = 64d$, and recalls what is recent: the signs of its newest d items are read with accuracy 0.80, those of its oldest eighth at chance.
- *Fuzzy addressing.* Identifying stored binary codes from noisy queries follows the exact binary-code law in all 20 settings (largest $|z|$ 2.44). With each query bit flipped independently with a fixed probability $0 \leq p < 1/2$, its cost per stored key tends to $\ln N/(\ln 2 - h(p))$ bits, set by the capacity $\ln 2 - h(p)$ of the binary symmetric channel. The registered slope test failed at two of its four flip rates: against a 10% tolerance, the measured slope ran 14.4% and 10.4% above capacity at $p = 0.05$ and 0.1. The registration itself had noted that the exact law's own slope runs 0–6% above capacity at these code lengths. Post hoc and unscored, the measured slopes sit 1.0 and 0.7 standard errors above the law's own at $p = 0.1$ and 0.2, within sampling noise, and 2.5 standard errors above it at $p = 0.05$, more than sampling noise readily explains. The law itself converges to capacity as the code length grows.
- *Addressing positions.* For a unary feature, the largest logit range needed to put a majority on a target, among the positions checked, grows like n^2 (fitted exponent 2.01, about $0.78(n-1)^2$). A rotary code (Su et al., 2021) needs 5.7–25.6 nats up to $n = 4{,}096$ with $2\log_2 n$ dimensions, within 3.1 times the floor $\ln(n-1)$. Recency carried in soft weights is unary whatever the code. With $K \geq 2$ an optimized rotary term orders 72–89% of the pairs that Proposition 11's staircase orders, at $K = 1$ almost none (at most 0.1%), and never more; a linear bias leaves 2.16–2.29 times as many pairs unordered as the staircase.
- *Each machine at its own wall.* On one recall task, the argmax fails at the code limit, with slope 0.526 in d after subtracting the logarithmic prefactor, against $C = 0.511$. A one-slot machine that reads the question first decides exactly as the argmax, by construction; what it saves is the state, one slot in place of the whole cache. Softmax at a fixed scale fails at its decoder limit, which moves 1.4–2.5 nats per doubling of τ. An outer-product state that reads the question last fails at its memory wall, near 30–57 items and 34–713 times below the code limit; the wall grows about linearly with d. A

wider value code moves only that wall: quadrupling it gains 0.21–0.24 nats, and the other machines' onsets stay put.

Recovered weak-evidence shortfalls. The accepted recovery's hard-argmax onsets are 8.372 and 18.006 nats at $c = 0.35, 0.5$. At $c = 0.5$, softmax $\tau = 120$, selection with $\tau = 20, M = 4096$, and the soft-final tree with $\tau = 60, k = 4$ have saved onsets 17.198, 16.476 and 16.157. Their nearest measured points are at $\ln N = 17.155, 16.462, 16.115$. The saved $(P_{\text{arg}}, S_{\text{arg}}, S_{\text{off}})$ triples there are $(0.582, 0.863, 0)$, $(0.650, 0.732, 0.079)$ and $(0.690, 0.734, 0)$: success is $P_{\text{arg}}S_{\text{arg}} + (1 - P_{\text{arg}})S_{\text{off}}$. Thus the selecting walker can succeed off the argmax, while failure even when the target is maximal contributes to every displayed shortfall. These are saved readouts, not new assessments or universal bounds (Appendix C.2).

4 Abstention depends on the null's test

A router that can retrieve must also be able to say "none of these". Whether it can do so at every length depends on how its null decision is made, not on what kind of machine makes it.

Proposition 3 (The test decides)**.**

(a) A competitive null. *A null of weight e^b that competes with the background sum $B = \sum_j e^{\tau s_j}$ has abstention mass $a = e^b/(e^b + B)$, and $a \geq \frac{1}{2}$ exactly when $e^b \geq B$. In mean field, $B = Nm(\tau)$, so it fails at $\ln N = b - \ln m(\tau)$. The schedule $b = b_0 + \ln N$ keeps the predicted a at $e^{b_0}/(e^{b_0} + m(\tau))$ for every N. But a matching key's weight is then $e^{\tau c}/\big(e^{\tau c} + N(e^{b_0} + m(\tau))\big)$, which vanishes, and retrieval fails at $\ln N = \tau c - \ln(e^{b_0} + m(\tau))$.*

(b) An absolute test. *A rule that abstains when the largest score is below a threshold θ abstains with probability $F(\theta)^N$ for independent background scores with continuous distribution function F. This is one half at $\ln N = \ln \ln 2 - \ln(-\ln F(\theta))$. If every background score lies below θ almost surely, it abstains at every N.*

(c) A sampled test. *A router that checks only M sampled items, given the row, finds no item above θ with probability $(1 - p_\theta)^M$, where p_θ is its routing mass above θ.*

The difference is what the decision sums over. Part (a) is the law of sums: a null that enters a normalization over all N items must grow like $\ln N$. A null that grows that way then costs retrieval, which is the two-sided law of Bruce (2026c) and Bruce (2026d). Part (b) reads a maximum. Its failure is set by the tail of the background at θ, not by a count, so it needs no schedule. It still fails where the background maximum reaches θ, which is the evidence wall in another form. Both parts read the background as exchangeable: (a)'s mean field takes B to grow like N, and (b) takes the background scores to be independent draws from a distribution that stays fixed as N grows. Positions break both in trained models. Inside the training length a rotary model's background sum grows like N^γ, with γ the measured growth, well below one for a fixed null, so by (a) a null that keeps abstention level needs only $\gamma \ln N$ (Bruce, 2026c). Beyond it, positions that are not interpolated raise the background maximum with length, and a maximum then fails as that maximum grows (Section 11). A finite number of competitive walkers realizes (a)'s decision as a binomial absorption event, which approaches the half-mass decision as the samples grow. A fixed number of verifying walkers, by (c), is not an exhaustive maximum: its abstention depends on its routing mass above θ, so it holds only while that mass stays small. In the constructed tests it held to 2^{30} items, though with weak evidence ($c = 0.5$) at $\tau = 30$ it was still falling there (Figure 6). On a present query its routing mass on the target falls with N, so it inherits the gain wall for retrieval. Appendix A.3 gives the proofs.

Constructed-key records. The package's abstention test calibrates each null at $N = 64$ on the product substrate (Figure 6). The recovery's recomputation of these records passes its registered comparison; the comparison that failed is in the gain-control test (Appendix C.2).

- *Competitive nulls.* The fixed softmax sink loses abstention within 0.43 nats of (a)'s onset in all 4 configurations. The walker's competitive killing loses it within 0.11 nats of the sink: the machines differ and the test is the same.

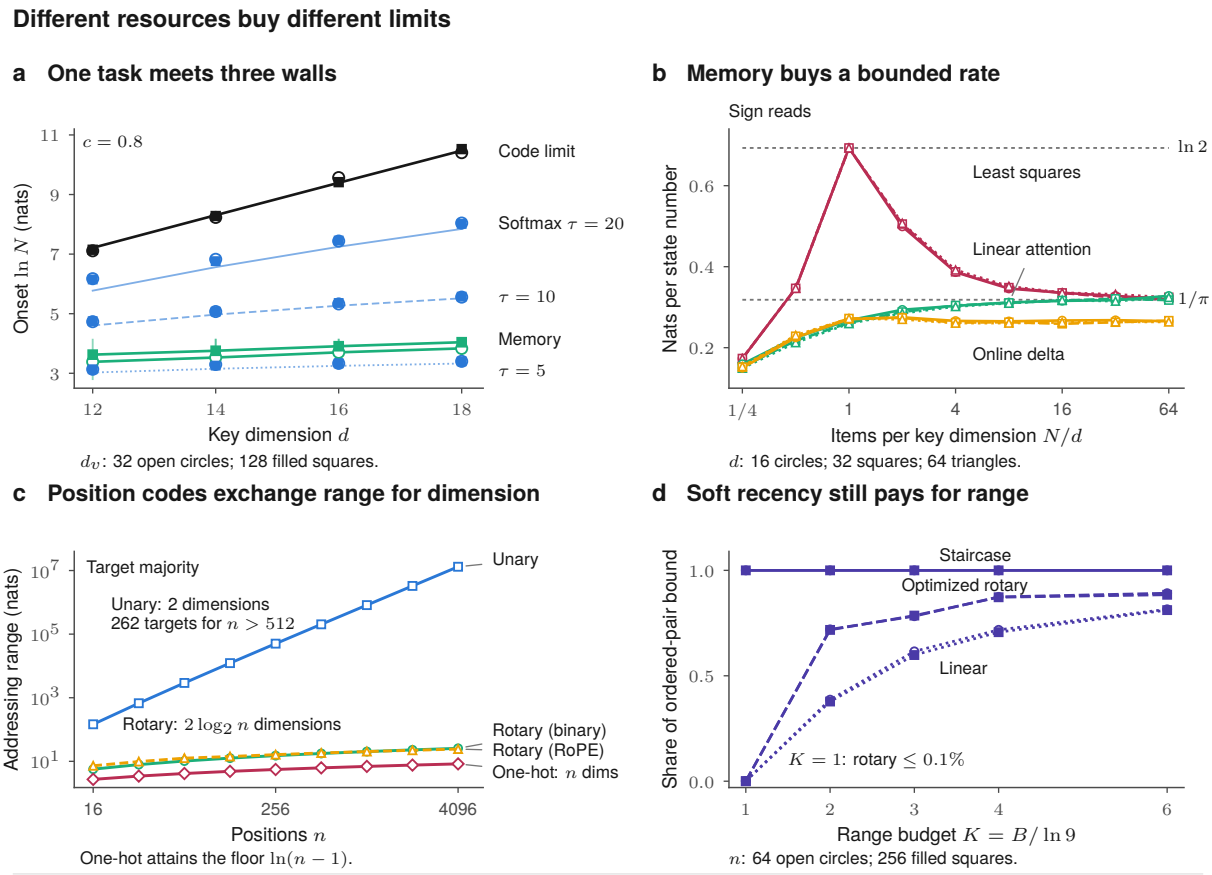


Figure 4: **Different resources buy different limits in three separate synthetic tests.** (a) The ledger-failure test, all eight settings: cache argmax (black), softmax (blue) and a question-last outer-product state (green), at match cosine $c = 0.8$ and 16 value classes. Open circles and filled squares give value widths $d_v = 32$ and 128; widening values moves only the memory wall. The question-first one-slot maximum equals the cache argmax in every trial, by construction. Argmax and softmax each use 400 trials per setting. Their saved onsets are $\ell = \ln \text{median}\, T$ for the first background index T; the common total-count axis plots $\ln(1 + e^{\ell})$, and bar ends are $\ln(1 + e^{\ell \pm s})$ for the saved bootstrap standard error s (2,000 resamples, seed 59). These are transformed intervals, not new standard errors of $\ln N$. The black line is the saved total-count code law; pale blue lines are the saved mean-field decoder references, reported rather than scored and least reliable at $\tau = 20$. The memory state uses its own 4,000 trials: its half-success crossing is interpolated in $\ln N$, with bars spanning the sampled sizes, not confidence intervals. (b) The memory test, all 81 memory cells: age-binned sign-read information divided by d^2, a lower bound on recalled information, with 8–16,384 trials per cell and no saved trial intervals. The three dimensions are separate curves. Linear attention reaches 0.315–0.328 from $N = 16d$; online delta peaks at 0.272–0.275 and settles at 0.264–0.267 by $N = 64d$. (c) The positional-code test, all 36 positional-addressing cells. Unary shows the maximum required range over checked targets: every target through $n = 512$; 262 targets each at $n = 1{,}024$, 2,048 and 4,096 (every $(n/256)$-th zero-based target, plus $0, 1, 2, n-3, n-2, n-1$ and $\lfloor (n-1)/2 \rfloor$). The latter are sampled maxima, not all-target certificates. Rotary solutions use shared amplitudes and are constrained and checked at every target; the one-hot code attains the exact floor $\ln(n-1)$. (d) Its separate 30-cell recency test: recorded ordered-pair counts divided by the finite-n bound of Proposition 11, at ratio $r = 9$ and range $K \ln 9$. Rotary points are the best saved solutions from 32 restarts, not certified optima: 72–89% of the bound for $K \geq 2$, at most 0.1% at $K = 1$. Lines join tested settings. The footer keeps the memory test's registered failure of its fuzzy-addressing slope check; no slope is refitted here. Sources: the three tests' vendored records (Table 11); no experiment, optimization, bootstrap or assessment was rerun.

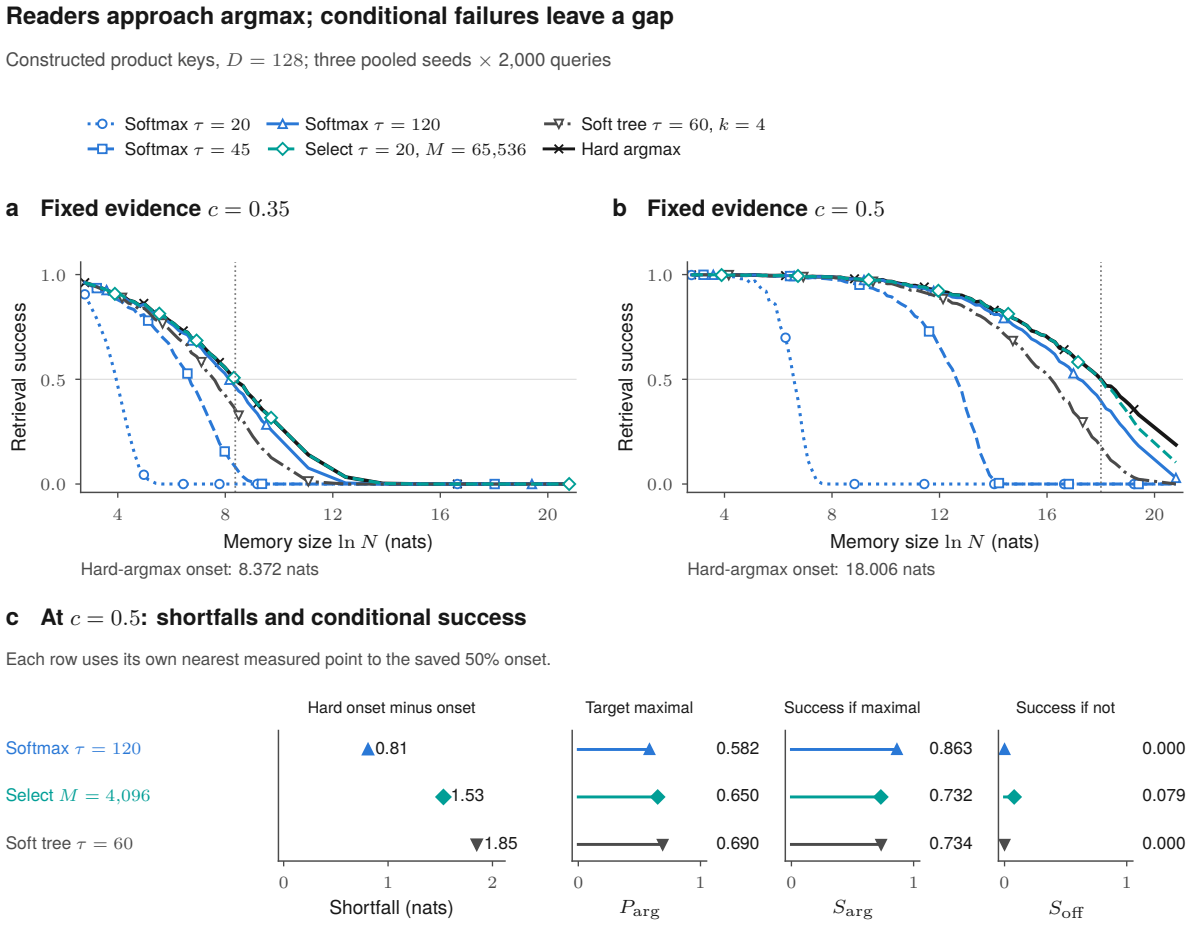


Figure 5: **Recovered readers approach the hard-argmax reference, with conditional failures explaining the remaining shortfall.** Constructed product keys have two 64-dimensional halves ($D = 128$). (a,b) All saved fine-grid points, merged with retained coarse points, at $c = 0.35, 0.5$: softmax $\tau = 20, 45, 120$; selection at $\tau = 20, M = 65{,}536$; a hard top-four shortlist per half with a soft final read at $\tau = 60$; and hard argmax. The vertical references are saved hard-argmax 50% onsets, 8.372 and 18.006 nats. Each curve pools three seeds with 2,000 queries each; softmax and soft-tree success require half the target mass, and walker probabilities are exact conditional on each query. (c) At $c = 0.5$, the three named readers use softmax $\tau = 120$, selection $\tau = 20, M = 4{,}096$, and soft tree $\tau = 60, k = 4$. Left: signed difference between the saved hard and machine onsets. Right: P_{arg} is the probability the target is maximal, and $S_{\text{arg}}, S_{\text{off}}$ are success conditional on that event and its complement. These probabilities are saved at the nearest measured point to each onset, not interpolated; conditional counts are 3494/2506, 3898/2102 and 4142/1858, respectively. No seed or paired-query intervals were retained. Pooled excesses are not clamped; these fixed-c tests are outside the premises of Proposition 1(b). The historical prediction that these readers converge on the wall remains partly met. Source: the recomputed constructed tests (Table 11; Appendix C.2).

- *The scheduled null.* The sink $+\ln N$ keeps abstention to 2^{30} items with exact queries, and loses retrieval instead: at $\ln N = 6.24$ and 10.40 for $\tau = 10$ and 20, against 6.17 and 10.52 from (a) with this substrate's background sum in place of $Nm(\tau)$, as for the gain wall in Section 3; the comparison values were derived after these records were read.
- *Absolute tests.* Verifying walkers and the tree's key check hold abstention to 2^{30} items with exact queries. With 64 walkers, retrieval fails at $\ln N = 13.94$ ($\tau = 10$), as (c) implies, while with $M = N$ walkers it does not.
- *Weak evidence ($c = 0.5$).* The exhaustive absolute test, the tree's key check, loses abstention first, at $\ln N = 14.68$ and 14.63 ($\tau = 20$ and 30), before retrieval at 17.98 and 18.08. The largest background cosine, averaged over queries, reaches the threshold near $\ln N = 14.57$ on the test's grid, which is where a test on the maximum fails. Product keys share their halves, so their scores are not independent and (b)'s formula does not apply as written. The tempered sink, which keeps both sides to 2^{30} items with exact queries, loses abstention at 9.77 and 13.58: tempering needs its margins (Bruce, 2026c).

A thought exercise behind the package claimed that only address-based routers keep abstention calibrated without a separate null schedule. The package's registration predicted that the claim fails as stated, and these tests confirm it: verifying walkers, a sampling machine, keep abstention too. What keeps abstention without a schedule, up to its own evidence wall, is a test on a per-item statistic, whatever machine runs it.

Recovered scheduled walkers. In the accepted recovery, competitive killing with a $\ln N$ schedule keeps abstention and loses retrieval at the product mean-field onset. The four saved retrieval crossings are 6.135, 10.488, 5.793 and 6.835 nats in the figure's row order; their registered predictions are 6.133, 10.488, 5.897 and 6.924. The registered checks on these walkers, of abstention and of the retrieval onset, pass. These are the recomputed records (Appendix C.2).

5 Tethers: consensus, coalescence and the price of concentrated routing

A machine that rebuilds every token from what it routes to it can lose the tokens themselves. Soft routing averages them toward one another. Hard routing copies one token onto another until whole groups share a state. A tether, a path back to each token's own input, prevents both, and what it costs depends on how concentrated the routing is.

Proposition 4 (The tether criterion)**.** *Let P be a fixed row-stochastic matrix and $0 \le \alpha < 1$. The tethered iteration $X_{t+1} = (1-\alpha)X_0 + \alpha P X_t$ has the unique fixed point RX_0, with $R = (1-\alpha)(I - \alpha P)^{-1}$, and it attracts every starting state.*

(a) If the rows of X_0 are unit vectors with Gram matrix G, nearest-cosine decoding returns token i's own identity, with no tied competitor, exactly when $(RG)_{ii} > (RG)_{ij}$ for every $j \neq i$. For orthonormal rows this is $R_{ii} > \max_{j \neq i} R_{ij}$.

(b) With orthonormal inputs and hard routing, $P_{i,f(i)} = 1$, a token on a cycle of length m keeps its identity for every $\alpha < 1$. A token $d \ge 1$ steps before a cycle of length m keeps it exactly when $\alpha^d + \alpha^m < 1$: $\alpha < (\sqrt{5}-1)/2 \approx 0.618$ for $(d, m) = (1, 2)$ and $\alpha < 1/\sqrt{2} \approx 0.707$ for $(d, m) = (2, 2)$.

(c) With orthonormal inputs, on a two-cycle, after t rounds from the token's own input, the token keeps its identity at every even t and at odd t exactly when $1 - \alpha > 2\alpha^{t+1}$.

R is the restart resolvent, the lottery operator of Bruce (2026a, §1.3): a token's output is its own input with weight $1-\alpha$, plus everything that returns to it along paths of routing. With P fixed, the iteration is the model of social influence of Friedkin & Johnsen (1990), in which every actor stays anchored to its initial opinion; its anchor-free limit $\alpha \to 1$ is the consensus model of DeGroot (1974). In looped networks the tether is input re-injection, or "recall", which, with progressive training, keeps them stable past their trained number of loops (Bansal et al., 2022; Yang et al., 2024a). A token's own weight is at least $1-\alpha$, and return paths raise it. Soft routing spreads its mass over many tokens, so the own weight wins easily.

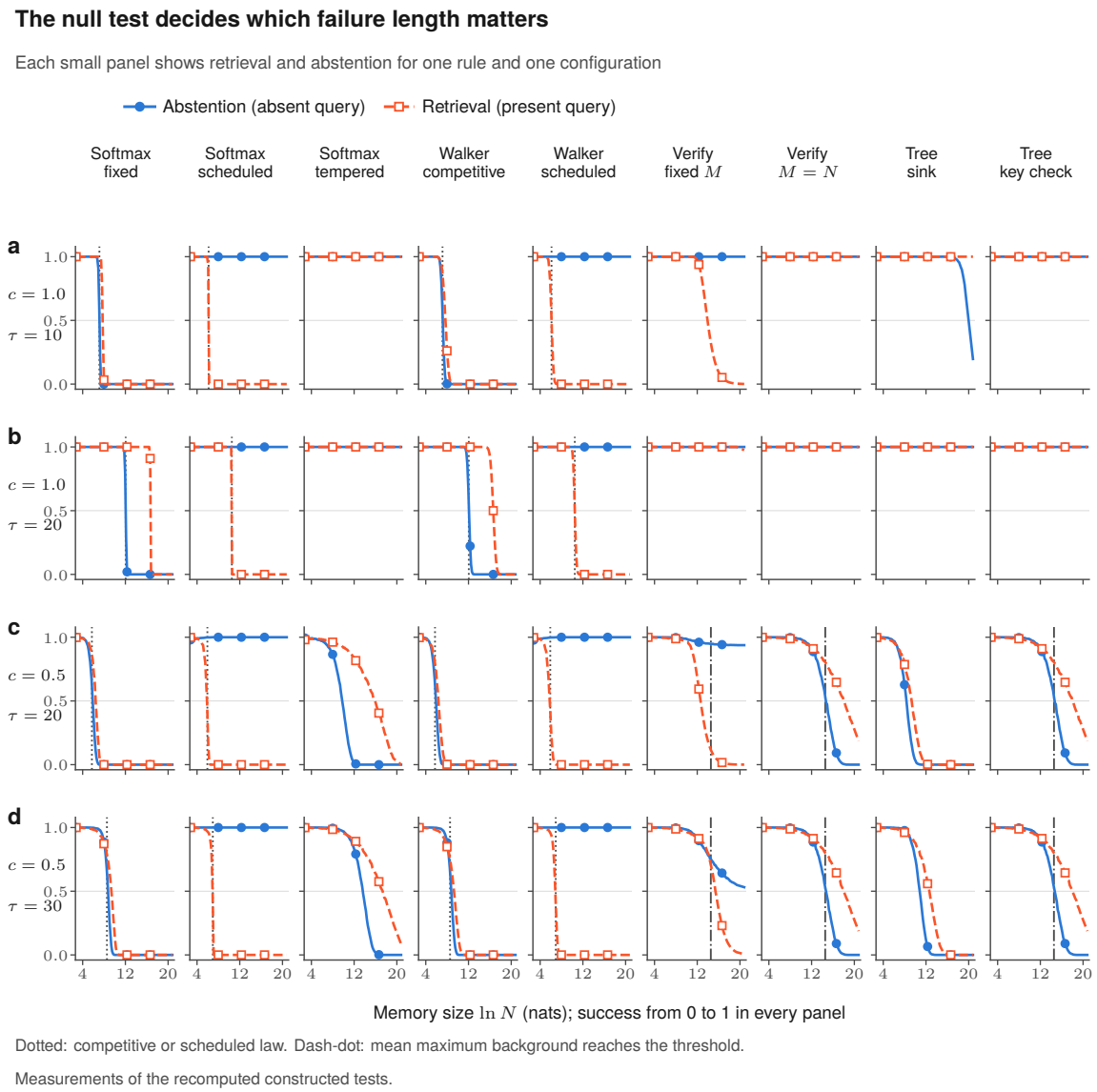


Figure 6: **A scheduled competitive null preserves abstention at a retrieval cost; absolute tests have a separate evidence limit.** Recovered product keys ($D = 128$), in four row configurations: (a,b) $c = 1$, $\tau = 10, 20$; (c,d) $c = 0.5$, $\tau = 20, 30$. All nine rule columns show absent-query abstention (blue filled circles, solid) and present-query retrieval (orange open squares, dashed). Nulls are calibrated at $N = 64$; walkers use $M = 64$ except the scaled verifier ($M = N$). Dotted lines show the fixed or scheduled competitive law; dash-dot lines show the saved mean-maximum threshold crossing, when finite. Tempering scales content and sink logits by $\ln N / \ln 64$; the tree sink competes over 4^2 shortlisted keys. Each point pools three seeds with 2,000 queries per type; conditional walker probabilities are exact. No seed intervals were retained. Source: the recomputed constructed tests (Table 11; Appendix C.2).

Hard routing concentrates it, so the criterion bites. The routing graph of a hard router is a functional graph: every token points to exactly one token, and the paths end in cycles. For a symmetric, strict, self-excluding nearest-neighbour score the cycles have length two, because a longer cycle would need its edge scores to rise all the way around it. Ties or asymmetric scores allow longer cycles, and self-routing allows cycles of length one. Tokens on a cycle keep their identity at every α. Tokens that feed a two-cycle need a stronger tether:

$\alpha < 0.618$ one step away and $\alpha < 0.707$ two steps away. These thresholds compare entries of R, so they decide identity only under (a)'s decoding premise, which is why (b) and (c) assume orthonormal inputs. Two identical inputs can never be told apart, and near-orthogonal inputs need a margin; Appendix A.4 gives a sufficient one in terms of the coherence $\max_{i \neq j} |G_{ij}|$.

Without a tether the two kinds of router fail differently. Soft routers reach consensus: every token converges to one state. Hard routers coalesce: tokens collapse onto their cycles while the population as a whole stays spread out. A collapse measure based on spread sees the first and misses the second. For machines that are not pure smoothers, the measure that sees both is identity retention, the fraction of tokens that still decode to themselves. The collapse ratio of Bruce (2026a, §2.3) is defined for soft averaging operators; a hard router lies outside that scope rather than contradicting it. Looped transformers that simulate Dijkstra's algorithm, breadth- and depth-first search and Kosaraju's algorithm route hard in this sense, and finite precision limits that simulation (Back de Luca & Fountoulakis, 2024).

Constructed-key records. The package's tether test routes 256 product-key tokens carrying 64-dimensional random cargo for 64 rounds, with three seeds (Figures 7 and 8). The recovery recomputes this test with the same identities and state counts, agreeing with the historical records to within 2.6×10^{-15} (Appendix C.2).

- *No tether.* Softmax identity is at most 0.012 by round 8, and every seed ends in a single state. The walker's spread stalls at its sampling-noise floor, about 0.12. The hard tree's identity equals its two-cycle fraction at even rounds, 0.64–0.77, and is zero at odd rounds, while its spread stays at least 0.997 with 164–198 distinct states. It has coalesced, and its spread does not show it.
- *With a tether.* At $\alpha = 0.9$, softmax keeps every token's identity from the second round on, and the walker keeps 0.83–0.94. The tree keeps exactly its two-cycle tokens: every token that feeds a cycle in this graph fails (b)'s condition at $\alpha = 0.9$. At $\alpha = 0.7$ it keeps 0.023–0.047 more than that. The criterion predicts these to be tokens two or more steps from their cycle, which it admits at this α, two steps away just below its threshold 0.707. The cargo is random and 64-dimensional, not orthonormal, so cross-talk decides some of them, and the criterion ignores it. At $\alpha = 0.5$ every tree token keeps its identity.

Recovered token classes. The accepted class-resolved records follow the thresholds in all three seeds: the depth-one threshold $(\sqrt{5}-1)/2$, the depth-two threshold $1/\sqrt{2}$ and the odd-round condition $1 - \alpha > 2\alpha^{t+1}$ of (c). All four registered checks on the classes pass. At $\alpha = 0.7$, 26 of the 46 deeper tokens retain identity; that is the recorded decode, not the larger count predicted by the orthogonal criterion. The saved Gaussian expectation describes finite-cargo cross-talk and is not an interval estimate. Figure 7 shows these classes.

6 Gain control: the price of adaptivity is uniqueness

When routing is rebuilt from the machine's own output, the loop has gain. If the gain beats the tether, fixed points can multiply, and the output can then depend on where the iteration started. Removing gain control therefore shows up as a loss of uniqueness. Consider the tethered attention map $F(X) = \beta X_0 + \alpha P(X) X$, with $\beta = 1 - \alpha$ and $P(X)$ the row softmax of $\tau\, x_i^\top x_j$ over fixed visible sets V_i of sizes m_i; self may be excluded, but the pattern does not change with X. Write $\|X\|_{\mathrm{r}} = \max_i \|x_i\|_2$ for the row radius of X, $\max_{i,j} \|x_i - x_j\|_2$ for its row diameter, and $R_0 = \|X_0\|_{\mathrm{r}}$.

Proposition 5 (Gain control)**.**

(a) The basic certificate. *The row ball $\|X\|_{\mathrm{r}} \leq R$ with $R \geq R_0$ is invariant, and on it $\mathrm{Lip}(F) \leq \alpha(1 + 2\tau R^2)$. If $\alpha(1 + 2\tau R^2) < 1$, the ball holds a unique fixed point, which attracts every start in it.*

(b) A smaller absorbing set. *Take any certified lower bound $0 < \ell_i \leq e^{-2\tau R_0^2}/m_i$ on the routing probabilities, and set $c_i = \beta x_{0i} + \alpha\beta\ell_i \sum_{j \in V_i} x_{0j}$ and $r_i = \alpha(1 - \beta m_i \ell_i) R_0$. The product of balls $\mathcal{K} = \prod_i \overline{B}(c_i, r_i)$ is invariant, lies inside the row ball of radius R_0, and contains every trajectory from that ball after two steps. So $\alpha(1 + 2\tau R_{\mathrm{eff}}^2) < 1$, with $R_{\mathrm{eff}} = \max_i(\|c_i\|_2 + r_i) \leq R_0$, certifies a unique fixed point in the original ball, attracting every start in it.*

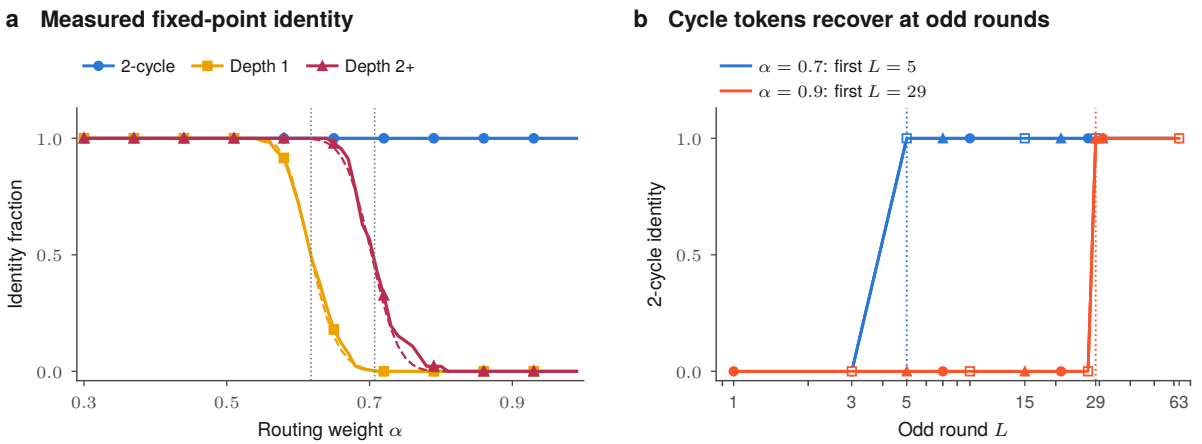


Figure 7: **Recovered identity measurements resolve token classes and the odd-round transient.** Frozen hard routing uses $N = 256$ product-key tokens, 64-dimensional random unit-vector cargo and three seeds. (a) Solid curves are saved pooled fixed-point identity fractions for 544 cycle, 178 depth-one and 46 deeper tokens across the seeds. Dashed curves show the saved Gaussian expectation $\sum_i p_i/n$, not uncertainty; deterministic ties with undefined expectation are omitted from that curve. Dotted references mark $(\sqrt{5}-1)/2$ and $1/\sqrt{2}$. (b) All three seeds at $\alpha = 0.7, 0.9$, with circle/square/triangle markers and no offsets; coincident paths are retained. The first identity-preserving odd rounds are $L = 5, 29$, as predicted by $1 - \alpha > 2\alpha^{L+1}$. Source: the constructed-test recovery's class tables and its saved odd-round checks (Table 11); all four registered checks on the classes pass. No confidence intervals are implied (Appendix C.2).

(c) A local certificate. *For a candidate $\widehat{X}$ with residual $e = \|F(\widehat{X}) - \widehat{X}\|_{\mathrm{r}}$, a radius $r > 0$, $R = \|\widehat{X}\|_{\mathrm{r}} + r$ and $q = \alpha(1 + 2\tau R^2)$: if $q < 1$ and $e \le (1 - q)r$, the ball of radius r around $\widehat{X}$ holds a unique fixed point, which attracts every start in that ball and lies within $e/(1 - q)$ of $\widehat{X}$. On a convex domain of row radius R and row diameter at most D, q may be replaced by $\alpha(1 + \tau RD)$.*

The certificates are sufficient, not necessary. A radius measured at a fixed point certifies nothing by itself. Part (b) builds an invariant set before any radius is substituted, and part (c) certifies only its own ball, and only with a verified residual (Appendix A.5). Nor can a smaller ball centred at zero be invariant in general: if every row is ru, with u the direction of an input row of norm R_0, that input's output row has norm $\beta R_0 + \alpha r > r$. For unit-norm inputs the basic certificate holds for $\tau < (1 - \alpha)/(2\alpha)$, a threshold that vanishes as $\alpha \to 1$. If the true onset of multistability stays bounded away from zero, the certificate's conservatism must therefore grow at least like $\alpha/(1 - \alpha)$. That is a conditional explanation of a growing gap, not a law for where the transition sits, and we leave its location open. The gain-control protection of Bruce (2026d) is the full-map form of this sufficient condition.

Retrieval against fixed patterns. When the routing reads fixed stored patterns instead of the machine's own states, as modern Hopfield memory does, the certificate has a sharp form. Anchored retrieval keeps a unique fixed point, and the query's identity, whenever $\alpha\tau D^2/4 \le 1$ for patterns of diameter D, and no smaller constant works (Proposition 18 in Appendix A.11).

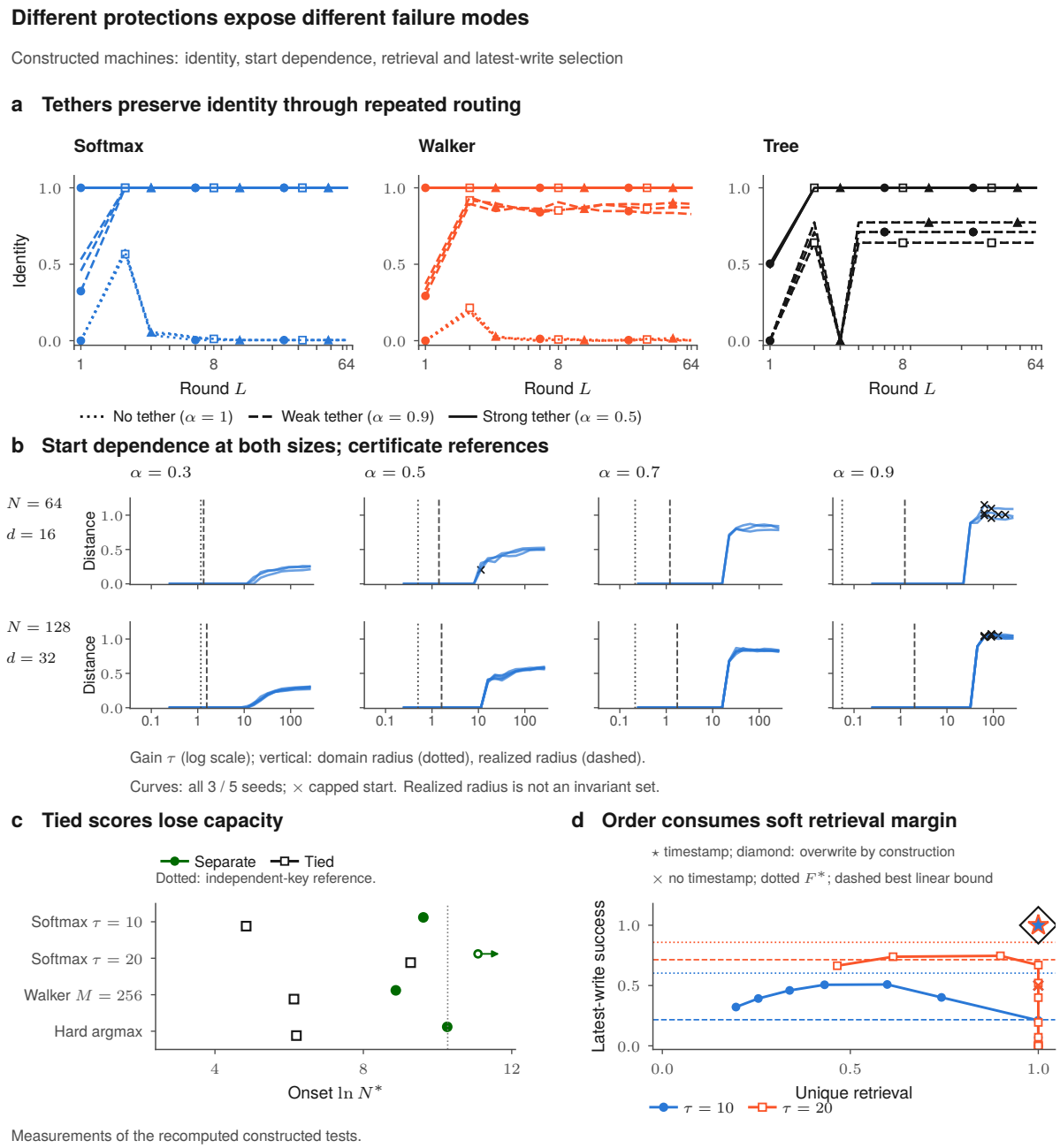


Figure 8: **Removing different protections exposes different failure modes in constructed machines.** (a) Identity over routing rounds: all three seeds at $\alpha = 1, 0.9, 0.5$; markers distinguish seeds. (b) Relative endpoint distance $\|Y - X\|_F / \|X\|_F$ for both sizes and every α, with all three/five seeds. Crosses mark either capped start. Vertical references are saved median tight certificates at the domain and realized radii; the latter is not an invariant set. (c) Saved tied/untied retrieval onsets; the arrow is right censoring, not an interval. The independent-key reference, 10.274 nats, is the argmax's evidence wall at $c = 0.35$, the cosine of the walker and hard-argmax rows. (d) All 12 recency biases at each gain, timestamp/no-timestamp walkers, overwrite by construction, and saved mean-field references. Source: the constructed-test recovery (Table 11). No seed intervals exist for the pooled retrieval curves. Appendix C.2 and Table 6 give counts, protocols and both certificate units.

Constructed-key records. The package's gain-control test iterates the map from two starting states on 64 unit-norm tokens in 16 dimensions, with self excluded and three seeds (Figure 8b). The two starts end within a relative Frobenius distance of 10^{-6} of each other, in every seed, up to $\tau = 11.3$, 8.0, 16.0 and 22.6 at $\alpha = 0.3$, 0.5, 0.7 and 0.9. There the basic certificate stops at $(1-\alpha)/(2\alpha) = 1.17$, 0.50, 0.21 and 0.06. It is conservative by a factor of about 9.6 at $\alpha = 0.3$, growing to about 400 at $\alpha = 0.9$ (16 and 74 between). The test's τ grid steps by factors of $\sqrt{2}$, so each factor, rounded down to two figures, is a lower bound within that step. These readings are provisional. The test ran below its registered size ($N = 128$, $d = 32$). 7 of its 252 cells did not converge within 1,500 iterations, two of them with an estimated local gain above one (1.08 and 1.07). And a second start's convergence was not recorded, so the readings show endpoint dependence, not a certified comparison of fixed points.

Recovered registered-size band. The accepted recovery includes the registered $N = 128, d = 32$ test with five seeds and a 5,000-update cap, retaining the 1,500-step checkpoint and both starts' convergence flags. Here τ_{loss} is the first grid value of τ at which the two starts no longer both converge to endpoints within 10^{-6} of each other with an estimated local gain below one. The test's saved medians of τ_{loss} over its seeds are 16.000, 16.000, 22.627 and 45.255 at $\alpha = 0.3, 0.5, 0.7, 0.9$. The registered check on this band passes: at every α the median lies within a factor of two of the historical median τ_{loss} (16, 11.31, 22.63 and 32). Figure 8b shows all seeds at both sizes, and Table 6 retains every per-seed readout and saved median/range, with the τ ratios and Lipschitz values separately named at both radii. Even at the realized radius the sufficient bound does not certify the observed uniqueness, as the registration predicted for the registered size; a usable certificate needs an invariant set, not the fixed point's radius (Appendix C.2).

7 Address and cargo

A routing machine carries two things: an address, which decides who talks to whom, and a cargo, which is what is said. They must travel in separate channels. If one channel carries both, the cargo becomes noise on the address. The general statement belongs to the protection lattice (Bruce, 2026d). Here we price the simplest violation, in the model the package's separation test uses. One channel carries both the address and the cargo: item j of class y_j scores $(z_j + g_{y_j})/2$, where z_j is its address's cosine with the query and g_{y_j} is the query's cosine with the cargo of class y_j. The address therefore arrives at half its amplitude.

Soft routers lose gain. For softmax, halving the address amplitude is the same as halving τ, and the cargo channel adds a background of its own. The mean-field gain wall falls from $\tau c - \ln m(\tau)$ to $\tau c/2 - 2\ln m(\tau/2)$. The separation test (isotropic keys, $D = 128$, exact queries) samples $\ln N$ in steps of 1.4 nats, and each mean-field onset falls inside the step where the measured success crosses one half. Its records are the package's historical constructed-key records; the recovery recomputed them to within 3.6×10^{-15} (Appendix C.2). Its measured onsets are read on this grid; on the recovery's finer grid, plotted in Figure 8c, they differ slightly. The tied router fails between $\ln N = 4.16$ and 5.55 at $\tau = 10$ and between 8.32 and 9.70 at $\tau = 20$, against mean-field onsets of 4.80 and 9.22. The untied router fails between 8.32 and 9.70 at $\tau = 10$, against 9.61, and at $\tau = 20$ not before the grid ends at $\ln N = 11.09$. At matched effective gain the tie costs only the cargo's background, $\ln m(\tau/2)$: 0.10 nats at $\tau = 10$. For a soft router, contamination is therefore mostly lost gain, which more gain can buy back where gain control allows (Section 6).

A hard router loses evidence. A router with no gain to lose takes the contamination in its evidence.

Theorem 6 (The contaminated argmax)**.** *Let the target's address score be c, let the other address scores be independent with continuous distribution function F, let C cargo scores $G_1, \dots, G_C$ be independent and independent of the addresses, and let each item belong to an independent, uniformly random class. Score item j by $(Z_j + G_{Y_j})/2$. With $F_g(x) = \frac{1}{C}\sum_{k=1}^{C} F(x - g_k)$, the target is the argmax of N items with probability*

$$p_N = \mathbb{E}_G\Big[\frac{1}{C}\sum_{y=1}^{C} F_G(c + G_y)^{N-1}\Big],$$

while without cargo it is $F(c)^{N-1}$. Unequal class probabilities replace both uniform averages by weighted sums.

Cargo is shared by all items of a class. Replacing it with fresh noise per item is a different model and need not give the same cost. The formula is the exact cost of contamination for an argmax. It does not say that the contaminated argmax is the best decision from contaminated scores. In the separation test ($D = 128$, 16 classes, $c = 0.35$ rather than the registered $c = 0.5$), a hard router fails at $\ln N = 10.27$ with clean scores and at 6.19 with tied ones, a ratio of 0.60. Walkers that select, with $M = 256$ or 65,536 samples at $\tau = 10$ or 20, show ratios of 0.60–0.67. Without cargo, the theorem gives $\ln N = 10.27$ in closed form; integrating its conditional law numerically gives 6.16 with cargo, a ratio of 0.59980, and the integrated law's largest gap from the measured tied curve is 0.019 in probability. That ratio is a numerical evaluation for this model, from six scrambled quasi-Monte Carlo replicates, not a universal constant or a rigorous enclosure (Appendix A.7).

Factorized addresses are immune. Separation can be built in. Give the addresses product structure, so that item (i, j) scores $a_i + b_j$.

Proposition 7 (Clean factorized addresses)**.** *The global argmaxes of $a_i + b_j$ are the pairs of half-argmaxes, and a Cartesian top-k shortlist built from the two clean half-scores contains one of them for every $k \geq 1$, and the global argmax itself when it is unique. If routing and final selection use only those scores, no change in the cargo changes the selection. If the final selection instead scores $a_i + b_j + v_{ij}$, the clean winner (i_*, j_*) stays the unique winner exactly when $v_{ij} - v_{i_*j_*} < a_{i_*} + b_{j_*} - a_i - b_j$ for every other shortlisted pair, and stays a winner exactly when these inequalities hold with $\leq$.*

Immunity holds by construction of the channels. It does not make a noisy address find its target: inclusion in the shortlist still depends on the address's own evidence, and Appendix A.7 gives its exact conditional law for random codebooks. On tied scores the separation test's flat hard router finds the target only 42–54% of the time at $c = 0.5$, less often with 64 codewords per half than with 32. In the recovery's readout of the same draws (Appendix C.2), per-half search on clean codebooks finds the target 99.3% and 97.9% of the time with 32 and 64 codewords per half, at every tested shortlist size from $k = 1$ to 16: exactly as often as the clean global argmax is the target. At $c = 1$ it always finds it. Rescoring the $k \times k$ shortlist with tied scores lets the cargo back in, and more of it as the shortlist grows: the target is found 72.3% and 66.2% of the time at $k = 2$, and 53.7% and 43.1% at $k = 16$, close to the flat router on tied scores.

8 The arrow is free for hard machines and has a price for soft ones

The protection lattice's arrow is direction along a sequence (Bruce, 2026d). For a machine that reads a store written over time, direction is write order: such a machine often needs the latest write, not just a matching one, and so needs a way to tell later writes from earlier ones.

Order blindness. A soft router's weights are functions of its scores. Two identical writes get identical scores, so without a positional term the router splits its mass evenly between them and cannot tell which came first. A hard router that reads only the scores picks one of the two arbitrarily.

Hard arrows are free. A machine that compares timestamps selects the later write, and one that overwrites on write keeps only the latest. Neither adds anything to the scores, so neither spends any of the router's margin (Proposition 11(d)). They pay elsewhere. Timestamps cost $N \ln N \geq \ln N!$ nats of storage, and overwrite answers only latest-value queries, so it gives up the history (Proposition 14(e), Remark 16).

Soft arrows are priced. A soft router can order writes only through a positional term added to its logits. Say that it orders two identical writes with ratio $r > 1$ when the later gets at least r times the earlier's weight; since the two share one normalizer, that needs their positional terms to differ by at least $\ln r$. Proposition 11, proved in Appendix A.8, prices this in three parts.

- *Range.* With range B, at most $t(n, K + 1)$ of the pairs among n positions can be ordered, where $K = \lfloor B/\ln r \rfloor$ and $t(n, h)$ counts the edges of the Turán graph (Turán, 1941); a staircase of $K + 1$ near-equal blocks attains it. The ordered fraction tends to $K/(K + 1)$ as n grows.

- *Shape.* A linear bias λs, the form of ALiBi (Press et al., 2022), with positions s spread uniformly over $[0, 1]$ so that its range is λ, orders only $(1 - \ln r/\lambda)_+^2$ of the pairs in the continuum limit. At large range it therefore leaves about twice as many unordered as the staircase does.
- *Margin.* In mean field, if every position must stay retrievable, the positional term b_j is paid from the router's margin Δ above its gain wall: averaged over the positions j, $e^{b_j - \min b}$ can be at most e^{Δ}, though its range B may exceed Δ. At the wall nothing can be ordered, and the best ordered fraction $F^*(\Delta)$ stays below one for every finite margin.

In the constructed arrow test ($N = 1024$, $D = 128$, $r = 9$), $F^* = 0.60$ at $\tau = 10$ and 0.86 at $\tau = 20$ (Table 5). So, in mean field, no positional term of any shape reaches 0.9 latest-write success while every position stays retrievable. Recency carried in soft weights is unary whatever the positional code: an optimized rotary term never exceeds the staircase's bound (Section 3).

Constructed-key records. The package's arrow test shows each piece (Figure 8d). The recovery of the constructed tests recomputed this test with no difference in any recorded field; its registration reports these values as historical records (Appendix C.2).

Without a positional term, softmax splits the two copies evenly and fails every latest-write query, and a walker without timestamps returns either copy half the time. A walker that compares timestamps succeeds every time, and an address book that overwrites on write succeeds by construction. A linear recency bias of slope λ gives latest-write success that follows $(1 - \ln 9/\lambda)_+^2$ to within 0.02 up to the slope that maximizes it, and then falls as recent distractors gain. Its best on the tested slopes is 0.51 at $\tau = 10$ ($\lambda = 8$) and 0.75 at $\tau = 20$ ($\lambda = 16$). At those slopes the oldest quarter of single items is retrieved 0.00 and 0.60 of the time. No tested slope reaches 0.9 on latest writes.

Physical machines pay in entropy. For a stochastic physical machine whose routing is a Markov kernel with a positive stationary law, the arrow also has an energetic price. Split the stationary flow $Q_{ij} = \pi_i P_{ij}$ into its symmetric part S and its antisymmetric part A; this is the π-weighted split of Bruce (2026a, §3.2), not the plain symmetrization of P. If $|A_{ij}| < S_{ij}$ on every edge with positive flow, the entropy produced per step is $2\sum_{i,j} A_{ij}^2/S_{ij}$ plus nonnegative even terms (Theorem 13, after Schnakenberg, 1976). Circulation therefore costs entropy at second order in the current, whatever its sign. This is a statement about a physical machine's Markov dynamics: the physical reading assumes a thermodynamic model and its units, and an attention matrix by itself is not a measured dissipation. The statement is checked numerically on a lopsided three-state chain and on 1,000 random three- and five-state chains with full support and currents at half their positivity limit. On every chain the entropy production is at least its quadratic term on every edge, it is even in the current to within 3×10^{-16}, and its relative excess over the quadratic term shrinks as the square of the current (fitted slopes 2.0006–2.0016 on the random chains and 2.004 on the three-state chain, against 2). The strongly nonlinear regime rests on the three-state chain alone: at full current its entropy production is 1.344 times the quadratic term, against at most 1.099 on the random chains.

9 A catalog of lattice machines

The seven conditions assume no particular machine, so they can be read off any system in which content decides who talks to whom. This section maps 43 families onto them: variants of attention, sampling and search machines, memories, recurrent and graph machines, physical substrates, and some biological and social systems. The catalog is not exhaustive. Each entry says how the family pays the conditions it meets and which proposition, if any, predicts how it fails; most entries also say where the writer sits. Table 3 groups the families by the currency in which they pay the routing budget, and Appendix D gives every entry with its sources.

How to read an entry. Each entry carries one of four labels. A *proposition* entry lies inside the class that a proposition is stated for, so the proposition predicts its failure. Many families lie outside these classes. The competitive null (Proposition 3(a)) and the expression ledger (Proposition 14(c)) are stated for weights proportional to e^s under one normalizer, the gain certificate (Proposition 5) for the tethered attention map, and the price of order (Proposition 11) for a positional term inside a shared normalizer. For a family outside a

class, the entry gives the mapping as a *reading*: it follows by the same computation, or it is an interpretation, and nothing in this paper tests it. Physical and biological systems are *hypotheses*: the cited works establish their science, and reading them as machines that route by content is ours. Markets and consensus protocols are *analogies*. A prediction that the cited works do not make is marked as ours.

What the catalog shows. Three patterns run through it. First, the two budgets of Section 3 sort the currencies. Soft routers pay the gain budget in logit scale, sampling machines pay it in samples through a select readout, and hard, addressed and indexed machines move it onto the query's evidence or onto address bits. No family buys evidence: every machine that ranks scores stops at the evidence wall at the latest (Proposition 1 and Proposition 14(b)). Second, the law of sums sorts the nulls (Proposition 3). A null that enters a single normalizer over all N items is competitive and, against exchangeable backgrounds, needs a $\ln N$ schedule, as an attention sink does. A null expert or a null document in a retrieval cascade is competitive too, but it competes only with a fixed set of experts or a bounded shortlist of documents, not with all N items. A null that tests one statistic is absolute and needs no schedule: a key check, an empty bucket, a match line, a membership filter. Read as an analogy, the same law separates the outside option of a logit choice model, which sits in its normalizer, from a reserve price compared with the highest bid. Third, the word "anchor" could name three different things in these families, and only one is a tether in the sense of Proposition 4. Re-injecting a token's own input, as a restart, a residual or a write-through does, is a tether. Decay toward zero, such as a decay gate on a recurrent state, is a contraction: it bounds a state but does not keep its identity. A starting point, such as the entry node of a graph search or the initial logits of capsule routing, is neither. The catalog keeps "tether" for the first.

Next tests. Nine families, starred in Table 3, are the most informative to test next, each in a trained machine.

- *Best-of-M verify-and-select*: whether a trained layer exchanges $\ln M$ for $\ln N$ (Proposition 2(a)) and keeps abstention with a sampled absolute test (Proposition 3(c)).
- *Block-routed attention*: whether block summaries carry the evidence (Proposition 1). We predict that it is robust to length with no temperature schedule while they do, and that it fails when the evidence is spread across blocks. The small trained tree of Section 10 is a block router of this kind: with no schedule it pays a length penalty beyond its training context, and whether its block summaries carried the evidence was not measured.
- *Stick-breaking attention*: a built-in null and a nearest-first arrow. We predict that it orders writes at no positional cost but needs a $-\ln N$ background bias to keep abstention at length.
- *Sinkhorn attention with dustbins*: two-sided conservation. We predict that, without a dustbin, irrelevant keys still receive mass, and that a dustbin sized to the other side keeps abstention at length with no schedule, while one of fixed capacity cannot.
- *Graph walkers*: whether a navigable graph brings the walker's proposal down to about $\log N$ hops, and how far below the evidence wall its search misses put its own wall.
- *Mixture-of-experts routers*: routing collapse, which we read as the router's runaway. It feeds back through training, so it lies outside the class of Proposition 5.
- *Vector-quantized memory*: whether learned codes pay the address bits that identification costs (Proposition 14(a)). We predict that codebook collapse is the address book's runaway.
- *Modern Hopfield memory with an anchor*: whether a trained layer that re-injects the query at every step stays inside the certified region of Proposition 18, where it keeps the query's identity and a unique fixed point, and what it retrieves outside it. We predict that without the anchor the iteration collapses queries onto stored patterns.
- *Walker plus tree*: the three-way split of the budget among the tree's evidence, samples and gain.

Several catalog families also train end to end as the token mixer of otherwise identical small models (Section 10), but this paper reports none of these nine tests.

Group	How it pays the routing budget	Members	Evidence
Soft routers that compete for a conserved mass	Gain: a logit scale, and a $\ln N$ schedule for a null that competes with all N items under one normalizer (Propositions 2 and 3)	Softmax, sigmoid, stick-breaking* and Sinkhorn* attention; slot attention; capsule routing	Propositions for softmax; readings for the rest
Stochastic and sampling machines	Samples: when no sample outranks the target, a select readout buys a nat of $\ln N$ per nat of $\ln M$, and an average buys none (Proposition 2)	Restart walkers; best-of-M verify-and-select*; graph walkers*; two hypotheses	Propositions for walkers and verify-and-select; restart walkers measured in the constructed tests (Sections 3 and 4); hypotheses otherwise
Hard, hierarchical selection	Evidence: a hard stage that keeps k of N items moves $\ln(N/k)$ onto the query's evidence, and a soft final stage pays $\ln k$ of gain	Product-key trees; block-routed attention*; indexer–reranker cascades; mixture-of-experts routers*; mixture of depths; sorting networks	Propositions; trees measured in the constructed tests (Sections 3 and 7); readings for depth routing and sorting
Addressing and memory machines	Address bits or dimension: naming one of N items costs about $\ln N$ nats however it is paid (Proposition 14(a))	Address books; locality-sensitive-hashing attention; vector-quantized memory*; Bloom filters; holographic memory; modern Hopfield memory*; external nearest-neighbour memory; content-addressable memory	Propositions; addressing measured in the constructed tests (Section 3), overwrite by construction (Section 8); a reading for content-addressable memory
Physical and analog machines	Dynamic range and energy: one competitive stage whose weights span a ratio of at most R buys about $\ln R$ nats (Proposition 19), and circulation costs entropy (Theorem 13)	Relaxation networks	Hypotheses; two propositions about ideal circuits (Appendix B)
Chemical, biological and social machines	Varied: routing tables and hop limits; bids and reserve prices; selection and amplification	Packet-switched networks; hypotheses and analogies	A reading; hypotheses; analogies
Recurrent machines with bounded state	A compressed state, which caps recall (Proposition 14(d)), or steps of local routing	Gated linear attention and state-space models; the delta rule; test-time-training layers; neural cellular automata	Propositions for memory, measured for linear and delta-rule states (Section 3); readings otherwise
Mixers that do not depend on content	None: a frozen operator routes nothing by content	Wavelet and multiresolution mixers	Outside the propositions
Graph and geometric machines	Neighbour selection, precision, or a bounded state	Message passing on learned nearest-neighbour graphs; hyperbolic embeddings; random-feature attention	Propositions; a reading for hyperbolic embeddings
Composite machines	Split across stages: an index's evidence, samples and gain; in hybrids, a bounded state plus the cache of a few attention layers	Hybrids of recurrence and attention; retrieval-augmented cascades; walker plus tree*	Propositions

Table 3: The catalog by group. Each group pays the routing budget in its own currency. Starred families ($\star$) are the next tests named in the text. Appendix D lists every family with its sources, how it meets the conditions, the proposition, if any, that predicts its failure, and its label (proposition, reading, hypothesis or analogy); the other hypotheses and the analogies are listed only there.

10 Small trained tests

The propositions describe machines with given scores. A design space is useful only if its machines can also learn their scores, so the first trained question is a plain one: do the machines train? Seven families were trained end to end as the token mixer of otherwise identical small pre-normalized models, beside softmax. A walker samples M keys in proportion to their softmax weights and renormalizes over what it lands on. A block router, labelled *tree* in the figure and tables, routes each query to its top blocks, chosen by their mean key, and runs a softmax over them and the query's own block. An address book hashes keys into buckets with random rotations and competes over the buckets its query matches, plus its own token. A relaxation network computes a truncated restart resolvent (Section 5) inside the layer. A market balances rows and columns (Sinkhorn's iteration, images only). A binding rule scores each key with an independent sigmoid offset by $-\ln n$, where n counts the keys visible to the query (language only). A bounded state keeps a linear-attention memory, with a learned decay on the language model. The tasks are deliberately small: a vision transformer on CIFAR-10 at 32 pixels, with about 141k parameters, trained for 1,170 steps with five seeds, and a byte-level language model on WikiText-2 with a 128-byte context, with about 594k parameters, trained for 1,500 steps with ten seeds. Everything except the mixer is held fixed, and the models differ in size by at most 0.03%. Compute is not matched, and no speed claim is made. The test was registered before it ran (Appendix C.5 gives the protocol, and Tables 7 to 10 the saved estimates and prediction outcomes).

Every machine trains. The accuracies, losses and contrasts below are seed means. Image accuracy at 32 pixels ranges from 0.549 (the relaxation network) to 0.609 (the market), against 0.602 for softmax. The language models reach 2.14 to 2.28 bits per byte at the training context, against 2.24 for softmax. Ten of the twelve informative pairs land within the registered parity band of softmax, 5 points of accuracy on images and 0.10 bits per byte on the language model. The two exceptions sit on either side of it: the relaxation network is 5.3 points below softmax on images, and the bounded state is 0.100 bits per byte better than softmax on the language model, just past the band.

Length transfer differs by machine. The language models trained on 128-byte windows and were evaluated on 512 bytes. The paired transfer Δ_{512} compares the loss on positions 128 to 511 with the full context against the loss when the same text is cut into 128-byte chunks. It is a net quantity: a machine that uses the longer context well has $\Delta_{512} < 0$, and $\Delta_{512} > 0$ means that what the longer context costs outweighs what the chunked pass loses at its resets. The three machines with an explicit length rule, a decay, a $-\ln n$ offset and an in-layer restart, show no penalty: Δ_{512} is -0.0327 for the bounded state, -0.0070 for the relaxation network and -0.0025 for the binding rule, in bits per byte over ten seeds. The other four machines all pay one: $+0.0053$ for the address book, $+0.0102$ for softmax, $+0.0110$ for the tree and $+0.0198$ for the walker, each with both 95% intervals above zero (Figure 9c).

The walker degrades most, and its paired contrast with every other machine is above zero, as Proposition 2 suggests: with M fixed, its budget, gain plus $\ln M$, does not grow with N, although the proposition's independent samples and select readout only approximate the trained walker, which samples without replacement and renormalizes. Letting M grow in proportion to the context is an evaluation rule that changes nothing inside the training length, so it runs on the same checkpoints. It cuts the walker's Δ_{512} by 0.0126, to 0.0073, below the tree and softmax.

A trained log-N scale does not remove the penalty. The registration made one structural prediction: that training with a log-N logit scale, learned per head (Nakanishi, 2025), would remove the competition machines' penalty, so that length transfer would split the machines by whether they carry a length rule. It did not. Softmax's Δ_{512} rose to 0.0131, a paired increase of 0.0029 (95% bootstrap interval 0.0020 to 0.0039), and the tree's rose to 0.0202. The walker's fell by 0.0010, which its intervals do not resolve. The scale trained without harm: it lowered the loss at the training context by about 0.003 bits per byte for all three machines. By the registered rule the structural reading is refuted, so the transfer is reported here machine by machine. Scaling the logits by the log-N ratio only at evaluation does not help either: it raises Δ_{512} for every logit-based machine except the address book, by 0.0022 bits per byte for softmax (95% bootstrap interval 0.0017 to 0.0027) and by 0.0054 for the tree.

A second registered comparison shows how much of the split comes from the chunked pass. It compares the full context with a sliding window that always sees the last 97 to 128 bytes, so it has no resets. Against that window, every machine except the bounded state does worse with the full context: by 0.0113 bits per byte for the relaxation network, 0.0159 for the binding rule and 0.0220 to 0.0307 for the four competition machines. The bounded state, whose decay forgets, is the only machine that context beyond the training length leaves unharmed (0.00003).

Resolution transfer. From 32 to 48 pixels every image mixer loses accuracy, by 3.4 (the relaxation network) to 5.7 points (the bounded state), on seed means. Scaling the logits by $\ln 145/\ln 65$ at evaluation recovers 0.26 to 0.66 points in five of the six logit-based mixers and costs the tree 0.03. Only the walker's gain clears both intervals.

What the small tests show. Every machine trains, most of them close to softmax, so at this scale trainability does not single out softmax. The walker behaves as its proposition suggests: with a fixed sample budget it pays the largest length penalty, and letting the samples grow with N removes most of it. The structural reading of length transfer does not survive: a log-N scale learned in training does not remove the penalty of the competition machines it was trained with at this scale, and it raises it for softmax and the tree. These are small models, short contexts and one data set per task. They show that the machines train and how they transfer, not how they rank at scale (Section 12).

11 Where trained machines sit relative to their walls

The propositions give each machine a set of walls: an evidence wall that no reader of the same scores passes (Proposition 1), a gain wall that a soft reader must pay to move (Proposition 2), a tether criterion (Proposition 4) and a gain certificate (Proposition 5). Constructed tests check that the walls are where the propositions put them. Trained models raise two further questions. The first is where retrieval fails. If a trained head fails at its evidence wall, then, under the proposition's joint likelihood condition, only better representations can help it; if it fails below, its gain or its routing is the bottleneck, and a better router could still help. The second is whether a null that tests a maximum keeps abstention across lengths without a schedule, as Proposition 3(b) says it can. A readout battery asks the first question of the small trained models of Section 10, together with the tether and the certificate. A max-statistic head trained in 125M-parameter language models asks the second.

Stage A: the small trained models. The readout battery read the walls of Sections 3, 5 and 6 off the trained models of Section 10: 135 cells, 100 of them language models and 35 image models, with every gate passing (Table 4). By its registered rule the stage is inconclusive. No head qualified as a retrieval head in any language-model cell (mean target weight at least 0.3 at $N_{\mathrm{ref}} = 128$, the training context, with the threshold frozen before scoring), so the two retrieval predictions, that softmax fails at its evidence wall and that Proposition 1(c) predicts the argmax onset, have no decided head; that is a qualification finding, and it licenses no conclusion about evidence walls or gain. With each routing matrix frozen and iterated without a tether, hard-routed image mixers lose identity while spread stays high in 0.000 of heads, against 0.000 for softmax (tree, LSH and walker, 5 paired seeds each): there is no trained support for coalescence missed by spread, though the coalescence of Section 5, measured on constructed keys, is not contradicted. The gain readout asks whether a head, iterated with its measured residual-update share $\hat{\alpha}$ in LayerNorm coordinates with hard routing held fixed, would meet the sufficient certificate. The median trained head has $\hat{\alpha}(1 + 2\tau R^2)$ between 13.3 and 35.2 across the 14 applicable groups (the walker counted once for each of its two sampling rules; per-seed medians 12.4–39.2), against the prediction's band of 0.1, set well below the certificate's bound of 1, with measured $\hat{\alpha}$ medians of 0.13–0.16 in the language models and 0.33–0.38 on images: trained heads sit far outside the certified region in general, not only at retrieval heads. Because the certificate is sufficient only, and on constructed maps uniqueness persisted 9.6 to 400 times beyond it (Section 6), this does not show that the trained heads are multistable. The battery's later stages, which read the same walls off larger trained models, are not reported here.

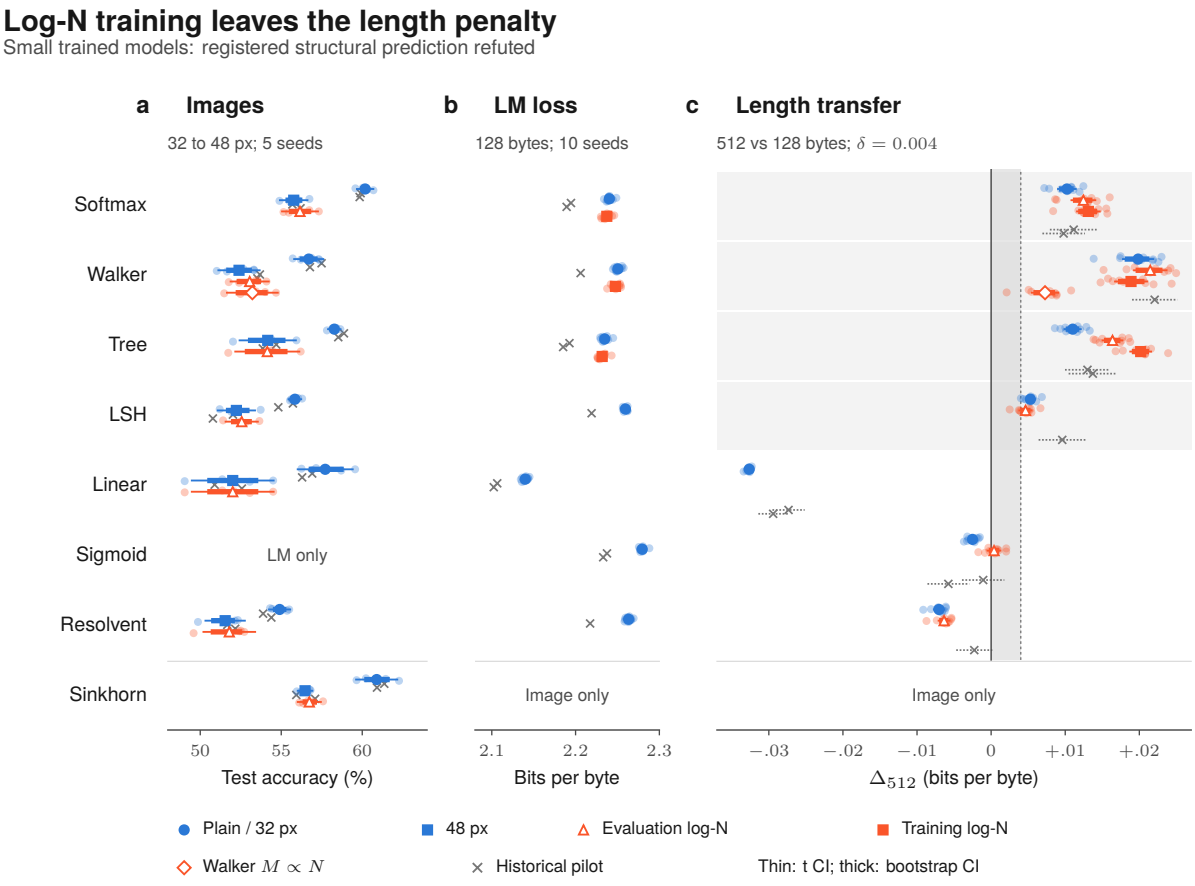


Figure 9: Training with log-N scaling leaves a positive length penalty for softmax, tree and walker, refuting the registered structural prediction. (a) CIFAR-10 accuracy at 32 and 48 px (five seeds, 10,000 test images). (b) WikiText-2 loss at 128 bytes (ten seeds). (c) Paired transfer Δ_{512}: full-context loss minus loss in 128-byte chunks, over positions 128–511 in 2,234 validation windows. Dots show seeds; large markers show means, with saved 95% Student-t (thin) and percentile-bootstrap (thick; 10,000 resamples) intervals. Orange marks alter the rule; diamonds increase the walker's samples with context. Gray rows mark the preregistered competition class; the vertical band is $[0, \delta]$, $\delta = 0.004$ bpb. These are prespecified groups, without a structural interpretation. Gray crosses are the 25 historical pilot records (one or two seeds; LM evaluated on 64 windows); dotted bars in (c) show their within-run ± 1 standard errors, never pooled with the registered run. Source: the small trained tests, first recovery run (Table 11); models have about 141k image or 594k LM parameters.

Registered prediction	Groups	Stage A	What decides it
Softmax retrieval fails at the evidence wall	Softmax language models, plain and with scalable softmax	inconclusive	no seed contributes a decided head (0 qualified retrieval heads)
The law of Proposition 1(c) predicts the argmax onset	As above	inconclusive	as above
Hard-routed image mixers coalesce where spread misses it	Tree, LSH and walker; 5 paired seeds	fails (all three)	spread-miss rate 0.000 against 0.000 for softmax; 95% CI [0, 0]
Trained τR^2 far below the certificate	14 groups (5 image, 9 language-model)	fails (14 of 14)	no seed has median $\kappa \leq 0.1$ (4 of 5 image or 8 of 10 language-model seeds required); per-seed medians 12.4–39.2

Table 4: Stage A of the readout battery on the small trained models of Section 10: 135 cells (100 language-model, 35 image), every gate passing. The registered verdict is inconclusive: the two retrieval predictions have no decided head. κ is the certificate's quantity $\hat{\alpha}(1 + 2\tau R^2)$ with the measured update share; the certificate needs $\kappa < 1$, and the prediction's band is 0.1. Generated from the vendored assessment (Table 11).

A max-statistic head in trained language models. The trained harness of 125M-parameter language models (Bruce, 2026c) trained the head of Figure 1 beside softmax, sink and gate rules at 1,024 tokens and evaluated every rule to eight times that length, with native positions and with positions interpolated by YaRN (Peng et al., 2024). With YaRN, the max-statistic head was the only rule whose null and match weights both changed by less than half a nat of log-odds from the training length to eight times it (-0.27 and -0.17 on seed means, jointly in 2 of 3 seeds), so the registered secondary prediction held on means. Five rules, softmax among them, kept the match weight alone better, but each lost more than half a nat on its null or first-token weight. Without YaRN it failed as the matching key fell far down the ranking and the background maximum grew with length: an evidence failure driven by positions beyond the training range, which break the exchangeability the theory's wall assumes, rather than by the number of keys. At this scale no head reached the harness's registered threshold for retrieval, so the result describes attention weights, not behaviour (Figure 10).

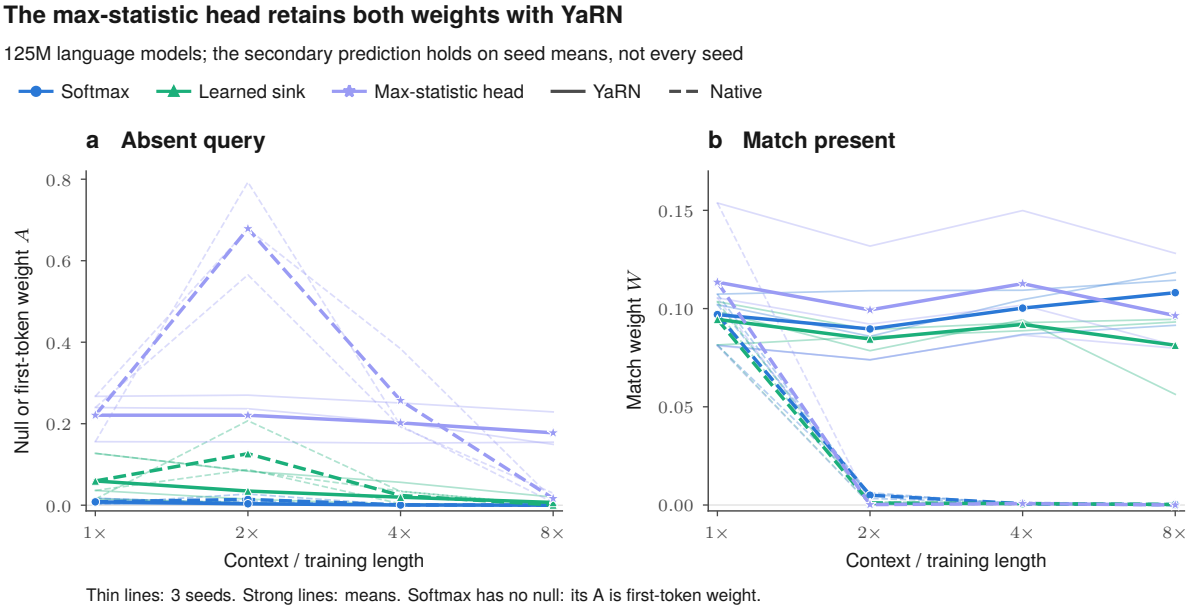


Figure 10: The max-statistic head retains both probe weights with YaRN, while native extrapolation fails. (a) Absent-query weight A: null mass for the learned sink and max-statistic head, first-token attention for softmax, which has no null. (b) Present-query match weight W. Solid lines use YaRN; dashed lines use native positions. Strong lines are seed means; thin lines show all 3 seeds, without a confidence interval. The models have 125M parameters and trained at 1,024 tokens. Each point averages 64 probes over 3 fallback heads; no head reached the retrieval-identification threshold, so these are attention weights, not behavioural retrieval. The registered secondary prediction holds on mean per-seed log-odds changes from training length to 8×: $\Delta \operatorname{logit} A = -0.27$ and $\Delta \operatorname{logit} W = -0.17$, both above -0.5, but it holds jointly in only 2 of 3 seeds. Natively, at 2× all scores fall, so the null weight rises and the match is lost; by 8× the background maximum has grown past its training value and the null collapses. The max head's rescue is due to positional interpolation with YaRN (Bruce, 2026c); it does not establish a guarantee beyond the tested lengths.

What the trained walls show. Neither question is settled. In the small language models no head met the probe's threshold for a retrieval head, so there is no retrieval failure to place against a wall, and the trained heads that do exist sit far outside the region the gain certificate covers, which, the certificate being sufficient only, does not make them multistable. At 125M the max-statistic head behaves as the law of sums predicts only when positions are interpolated with YaRN. Without interpolation, positions raise the background maximum with length, and a test on the maximum fails as that maximum reaches its threshold (Section 4).

12 The price list decides

The conditions price the protections, and the price list decides among the machines that pay them. Dense matrix multiplies and gradient descent favour softmax. Its log-sum-exp is smooth and convex in the scores, every protection is cheap on that hardware, and its gain and its null are one scalar each. Other price lists favour other points. Where memory dominates the cost, a bounded state pays the memory ledger in state and gets direction and recency by construction (Table 2). Sparse or event-driven hardware favours machines that touch few items, such as walkers and address books. Energy-bound analog hardware favours relaxation networks: with proportional shunts, a passive reciprocal circuit computes the tethered fixed point of a reversible routing kernel exactly (Proposition 20); a directed kernel needs non-reciprocal elements, and a device's dynamic range caps the gain of one soft stage (Proposition 19). These are readings under assumed prices, not measurements. This paper prices no machine on a common budget, and the lattice prices protections; it does not predict capability.

No price list buys evidence. The evidence wall belongs to the keys (Proposition 1). For random keys it grows with the dimension at the capacity of a Gaussian channel, and, for a uniformly placed target under the proposition's joint likelihood condition, every machine that ranks keys stops at it at the latest (Table 2). Gain, samples, trees and schedules buy back the decoder limit; only better keys move the code limit. So a machine's levers come in two kinds: those that change how it reads, which can bring it up to the wall, and those that change what it reads, which can move the wall.

The conditions also build. The law of sums gives a head the transformer lacks, a max-statistic head whose route and null both read maxima (Figure 1c). In trained 125M language models its null and match weights held up across eight times the training length, on the seed means, only with YaRN's positional interpolation (Section 11). Without interpolation the background maximum grew with length, and that maximum is exactly what the head's test reads. The price of order gives a second rule: a soft recency term should be a staircase rather than a linear bias, and exact order belongs to a hard mechanism, timestamps or overwrite, which spends none of the router's margin (Section 8).

The trained evidence so far is mixed. Every small machine trains, most of them close to softmax. Length transfer differs by machine, and how it divides the machines depends on the comparison. A log-N scale learned in training does not remove the penalty of the competition machines it was trained with; for softmax and the tree it raises it (Section 10). No head in the small language models met the probe's threshold for a retrieval head, so where their retrieval fails, at its evidence wall or below it, is still open (Section 11).

Limits. The constructed tests use synthetic keys: product-structured and isotropic keys with exact scores, and frozen or adaptive routing among 64 to 256 random tokens. Real keys are clustered. The propositions hold within declared classes, and a trained model need not sit inside them; the evidence proposition, for one, needs a joint condition on the likelihood of the scores that clustered keys can violate. The trained tests are small: models of under 600k parameters with short contexts, and 125M-parameter language models with three seeds for the max-statistic head. The physical, chemical and biological machines of Appendices B and D are readings off the lattice, and no device was simulated. The catalog maps machines from their published descriptions, checked against their sources; it measures none of them.

Next tests. Three tests come next. The first asks where each machine makes its features. Transport cannot write (Proposition 10), so every machine needs a writer, and where the writer sits is a design choice that the transformer does not use. A writer can sit where the machine already brings two items together, and a sparse machine brings few pairs together, so an encounter writer that reads both items of a pair becomes

affordable there. The hypotheses are that encounter writers pay only in sparse machines, at matched compute on relational tasks; that product-key trees need writers that derive factorized keys, without which per-level search misses the true argmax; and that address books need balanced codes, without which their code usage collapses below what their bits allow. The second is a ladder of tests at scale. Each machine that Section 9 marks as a next test, with its own writer, moves from constructed tests through synthetic trained tasks and small real data to language models of 125M to 350M parameters and vision transformers, at matched FLOPs and wall-clock time with kernels. It climbs a rung only with its protections verified: abstention calibrated across lengths, an anchor that holds identity and a certified gain. The third reads the same walls off larger trained models, and asks the max-statistic head's question again at a larger scale.

Until those tests run, the conditions say what any machine that replaces the transformer must still buy, and the price list says where it might win.

Use of AI tools

AI tools were used, under the author's direction, for several kinds of work: software development and debugging, mathematical derivation and checking, data analysis, literature search and citation checking, and drafting and editing of the text. The author reviewed and verified all AI-assisted content to the extent possible and takes responsibility for the paper.

References

Ryan Prescott Adams and Richard S. Zemel. Ranking via Sinkhorn propagation. arXiv preprint arXiv:1106.1925, 2011.

M. Ajtai, J. Komlós, and E. Szemerédi. An $O(n \log n)$ sorting network. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing (STOC '83)*, pp. 1–9. ACM Press, 1983. doi: 10.1145/800061.808726.

Habibullah Akbar. ATMA: Long-context language modeling via polar attention and gated-delta compression memory. arXiv preprint arXiv:2606.25156, 2026.

Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, James Zou, Atri Rudra, and Christopher Ré. Simple linear attention language models balance the recall-throughput tradeoff. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1763–1840. PMLR, 2024.

Artem Babenko and Victor Lempitsky. The inverted multi-index. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3069–3076. IEEE, 2012. doi: 10.1109/CVPR.2012.6248038.

Artur Back de Luca and Kimon Fountoulakis. Simulation of graph algorithms with looped transformers. In *International Conference on Machine Learning*, 2024.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Fred Baker. Requirements for IP version 4 routers. RFC 1812, RFC Editor, https://www.rfc-editor.org/rfc/rfc1812, June 1995.

Arpit Bansal, Avi Schwarzschild, Eitan Borgnia, Zeyad Emam, Furong Huang, Micah Goldblum, and Tom Goldstein. End-to-end algorithm synthesis with recurrent networks: Extrapolation without overthinking. In *Advances in Neural Information Processing Systems*, volume 35, pp. 20232–20242, 2022.

K. E. Batcher. Sorting networks and their applications. In *Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference (AFIPS '68 Spring)*, pp. 307–314. ACM Press, 1968. doi: 10.1145/1468075.1468121.

Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Peter W. Battaglia et al. Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261, 2018.

Assaf Ben-Kish, Itamar Zimerman, Shady Abu-Hussein, Nadav Cohen, Amir Globerson, Lior Wolf, and Raja Giryes. DeciMamba: Exploring the length extrapolation potential of Mamba. In *International Conference on Learning Representations*, 2025.

Vincent-Pierre Berges, Barlas Oguz, Daniel Haziza, Wen-Tau Yih, Luke Zettlemoyer, and Gargi Ghosh. Memory layers at scale. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 3831–3842. PMLR, 2025.

Guo-qiang Bi and Mu-ming Poo. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *The Journal of Neuroscience*, 18(24): 10464–10472, 1998. doi: 10.1523/JNEUROSCI.18-24-10464.1998.

Lacramioara Bintu, Nicolas E. Buchler, Hernan G. Garcia, Ulrich Gerland, Terence Hwa, Jané Kondev, and Rob Phillips. Transcriptional regulation by the numbers: models. *Current Opinion in Genetics & Development*, 15(2):116–124, 2005. doi: 10.1016/j.gde.2005.02.007.

Burton H. Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970. doi: 10.1145/362686.362692.

Sebastian Borgeaud et al. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2206–2240. PMLR, 2022.

Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006. doi: 10.1109/TIT.2006.874516.

Andrei Broder and Michael Mitzenmacher. Network applications of Bloom filters: A survey. *Internet Mathematics*, 1(4):485–509, 2004. doi: 10.1080/15427951.2004.10129096.

Neil D. B. Bruce. A spectral theory of token communication: Operators, amplification and the architecture of transformers. Zenodo, version 0.1, released with this paper, 2026a. URL https://doi.org/10.5281/zenodo.23541723.

Neil D. B. Bruce. Attention sinks at long context: One log-N law for retrieval and abstention. Zenodo, released with this paper, 2026c. URL https://doi.org/10.5281/zenodo.23541963.

Neil D. B. Bruce. The transformer as a constrained equilibrium: A lattice of protections for adaptive sequence mixing, and where the room is. Zenodo, released with this paper, 2026d. URL https://doi.org/10.5281/zenodo.23542051.

F. M. Burnet. A modification of Jerne's theory of antibody production using the concept of clonal selection. *Australian Journal of Science*, 20:67–69, 1957. Reprinted in CA: A Cancer Journal for Clinicians 26(2):119–121, 1976, doi:10.3322/canjclin.26.2.119; also reprinted at the end of Hodgkin, Heath and Baxter (2007).

F. M. Burnet. Immunological recognition of self. Nobel Lecture, 12 December 1960, https://www.nobelprize.org/uploads/2018/06/burnet-lecture.pdf, 1960.

Larry Carter, Robert Floyd, John Gill, George Markowsky, and Mark Wegman. Exact and approximate membership testers. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, pp. 59–65, 1978.

David Chiang and Peter Cholak. Overcoming a theoretical limitation of self-attention. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7654–7664. Association for Computational Linguistics, 2022.

Dah-Ming Chiu and Raj Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN Systems*, 17(1):1–14, 1989. doi: 10.1016/0169-7552(89)90019-6.

Eugene Choo and Aloysius Siow. Who marries whom and why. *Journal of Political Economy*, 114(1):175–201, 2006. doi: 10.1086/498585.

Krzysztof Choromanski et al. Rethinking attention with Performers. In *International Conference on Learning Representations*, 2021.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

Marco Cuturi, Olivier Teboul, and Jean-Philippe Vert. Differentiable ranking and sorting using optimal transport. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Damai Dai et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1280–1297. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.70.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision Transformers need registers. In *International Conference on Learning Representations*, 2024.

Mike Davies et al. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018. doi: 10.1109/MM.2018.112130359.

Soham De et al. Griffin: Mixing gated linear recurrences with local attention for efficient language models. arXiv preprint arXiv:2402.19427, 2024.

DeepSeek-AI. DeepSeek-V3.2: Pushing the frontier of open large language models. arXiv preprint arXiv:2512.02556, 2025a.

DeepSeek-AI. DeepSeek-V3.2-Exp: Boosting long-context efficiency with DeepSeek sparse attention. Technical report, https://github.com/deepseek-ai/DeepSeek-V3.2-Exp/blob/main/DeepSeek_V3_2.pdf, 2025b.

Stephen E. Deering and Robert M. Hinden. Internet Protocol, version 6 (IPv6) specification. RFC 8200 (STD 86), RFC Editor, https://www.rfc-editor.org/rfc/rfc8200, July 2017.

Morris H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974. doi: 10.1080/01621459.1974.10480137.

Mete Demircigil, Judith Heusel, Matthias Löwe, Sven Upgang, and Franck Vermet. On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168(2):288–299, 2017. doi: 10.1007/s10955-017-1806-y.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. arXiv preprint arXiv:2005.00341, 2020.

Peter G. Doyle and J. Laurie Snell. *Random Walks and Electric Networks*, volume 22 of *Carus Mathematical Monographs*. Mathematical Association of America, 1984. doi: 10.5948/UPO9781614440222. Revised edition (version 3.02, 2000) available as arXiv:math/0001057.

Wesley M. Eddy. Transmission Control Protocol (TCP). RFC 9293 (STD 7), RFC Editor, https://www.rfc-editor.org/rfc/rfc9293, August 2022.

David C. Fajgenbaum and Carl H. June. Cytokine storm. *New England Journal of Medicine*, 383(23): 2255–2273, 2020. doi: 10.1056/NEJMra2026131.

William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

G. David Forney, Jr. Exponential error bounds for erasure, list, and decision feedback schemes. *IEEE Transactions on Information Theory*, 14(2):206–220, 1968.

Noah E. Friedkin and Eugene C. Johnsen. Social influence and opinions. *Journal of Mathematical Sociology*, 15(3–4):193–206, 1990. doi: 10.1080/0022250X.1990.9990069.

Alfred Galichon and Bernard Salanié. Cupid's invisible hand: Social surplus and identification in matching models. *The Review of Economic Studies*, 89(5):2600–2629, 2022. doi: 10.1093/restud/rdab090.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized PageRank. In *International Conference on Learning Representations*, 2019.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495. Association for Computational Linguistics, 2021.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1263–1272. PMLR, 2017.

Siddharth Gollapudi, Nilesh Gupta, Prasann Singhal, and Sewon Min. Can language models actually retrieve in-context? Drowning in documents at million token scale. arXiv preprint arXiv:2607.01538, 2026.

Alex Graves. Adaptive computation time for recurrent neural networks. arXiv preprint arXiv:1603.08983, 2016.

Riccardo Grazzi, Julien Siems, Arber Zela, Jörg K. H. Franke, Frank Hutter, and Massimiliano Pontil. Unlocking state-tracking in linear RNNs through negative eigenvalues. In *International Conference on Learning Representations*, 2025.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Conference on Language Modeling*, 2024.

Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, and Nando de Freitas. Hyperbolic attention networks. In *International Conference on Learning Representations*, 2019.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: Retrieval-augmented language model pre-training. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3929–3938. PMLR, 2020.

Sariel Har-Peled, Piotr Indyk, and Rajeev Motwani. Approximate nearest neighbor: Towards removing the curse of dimensionality. *Theory of Computing*, 8:321–350, 2012. doi: 10.4086/toc.2012.v008a014.

Themistoklis Haris. kNN attention demystified: A theoretical exploration for scalable Transformers. In *International Conference on Learning Representations*, 2025.

A. V. Hill. The possible effects of the aggregation of the molecules of hæmoglobin on its dissociation curves. *The Journal of Physiology*, 40(Suppl.):iv–vii, 1910. doi: 10.1113/jphysiol.1910.sp001386. Proceedings of the Physiological Society, 22 January 1910.

Geoffrey E. Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with EM routing. In *International Conference on Learning Representations*, 2018.

Philip D. Hodgkin, William R. Heath, and Alan G. Baxter. The clonal selection theory: 50 years since the revolution. *Nature Immunology*, 8(10):1019–1026, 2007. doi: 10.1038/ni1007-1019.

John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the USA*, 79(8):2554–2558, 1982. doi: 10.1073/pnas.79.8.2554.

John J. Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences of the USA*, 81(10):3088–3092, 1984. doi: 10.1073/pnas.81.10.3088.

Daniele Ielmini and H.-S. Philip Wong. In-memory computing with resistive switching devices. *Nature Electronics*, 1(6):333–343, 2018. doi: 10.1038/s41928-018-0092-2.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing (STOC '98)*, pp. 604–613. ACM Press, 1998. doi: 10.1145/276698.276876.

Van Jacobson. Congestion avoidance and control. *ACM SIGCOMM Computer Communication Review*, 18 (4):314–329, 1988. doi: 10.1145/52325.52356.

Dirk Jalas et al. What is — and what is not — an optical isolator. *Nature Photonics*, 7(8):579–582, 2013. doi: 10.1038/nphoton.2013.185.

Pranav Jeevan and Amit Sethi. WaveMix: Resource-efficient token mixing for images. arXiv preprint arXiv:2203.03689, 2022.

Pranav Jeevan, Kavitha Viswanathan, Anandu A S, and Amit Sethi. WaveMix: A resource-efficient neural network for image analysis. arXiv preprint arXiv:2205.14375, 2024.

Glen Jeh and Jennifer Widom. Scaling personalized Web search. In *Proceedings of the Twelfth International Conference on World Wide Web*, pp. 271–279, Budapest, Hungary, 2003. ACM Press. doi: 10.1145/775152.775191.

Samy Jelassi, David Brandfonbrener, Sham M. Kakade, and Eran Malach. Repeat after me: Transformers are better than state space models at copying. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21502–21521. PMLR, 2024.

Peng Jin, Bo Zhu, Li Yuan, and Shuicheng Yan. MoE++: Accelerating mixture-of-experts methods with zero-computation experts. In *International Conference on Learning Representations*, 2025.

Ramesh Johari and John N. Tsitsiklis. Efficiency loss in a network resource allocation game. *Mathematics of Operations Research*, 29(3):407–435, 2004. doi: 10.1287/moor.1040.0091.

Vinay Joshi, Manuel Le Gallo, Simon Haefeli, Irem Boybat, S. R. Nandakumar, Christophe Piveteau, Martino Dazzi, Bipin Rajendran, Abu Sebastian, and Evangelos Eleftheriou. Accurate deep neural network inference using computational phase-change memory. *Nature Communications*, 11(1):2473, 2020. doi: 10.1038/s41467-020-16108-9.

Pentti Kanerva. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation*, 1(2):139–159, 2009. doi: 10.1007/s12559-009-9009-8.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive Transformers with linear attention. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5156–5165. PMLR, 2020.

Frank Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8 (1):33–37, 1997. doi: 10.1002/ett.4460080106.

Jack Kendall, Ross Pantone, Kalpana Manickavasagam, Yoshua Bengio, and Benjamin Scellier. Training end-to-end analog neural networks with Equilibrium Propagation. arXiv preprint arXiv:2006.01981, 2020.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*, 2020.

Kimi Team. Kimi Linear: An expressive, efficient attention architecture. arXiv preprint arXiv:2510.26692, 2025.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient Transformer. In *International Conference on Learning Representations*, 2020.

Ludger Klein, Bruno Kyewski, Paul M. Allen, and Kristin A. Hogquist. Positive and negative selection of the T cell repertoire: what thymocytes see (and don't see). *Nature Reviews Immunology*, 14(6):377–391, 2014. doi: 10.1038/nri3667.

Robert Kleinberg. Geographic routing using hyperbolic space. In *IEEE INFOCOM 2007 – 26th IEEE International Conference on Computer Communications*, pp. 1902–1909. IEEE, 2007. doi: 10.1109/INFCOM.2007.221.

Dmitry Krotov and John J. Hopfield. Dense associative memory for pattern recognition. In *Advances in Neural Information Processing Systems*, volume 29, pp. 1172–1180. Curran Associates, Inc., 2016.

Guillaume Lample, Alexandre Sablayrolles, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Large memory layers with product keys. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Adrian Łańcucki, Jan Chorowski, Guillaume Sanchez, Ricard Marxer, Nanxin Chen, Hans J. G. A. Dolfing, Sameer Khurana, Tanel Alumäe, and Antoine Laurent. Robust training of vector quantized bottleneck models. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2020. doi: 10.1109/IJCNN48605.2020.9207145.

Irving Langmuir. The adsorption of gases on plane surfaces of glass, mica and platinum. *Journal of the American Chemical Society*, 40(9):1361–1403, 1918. doi: 10.1021/ja02242a004.

Manuel Le Gallo and Abu Sebastian. An overview of phase-change memory device physics. *Journal of Physics D: Applied Physics*, 53(21):213002, 2020. doi: 10.1088/1361-6463/ab7794.

Kyle Lee, Corentin Delacour, Kevin Callahan-Coray, Kyle Jiang, Can Yaras, Samet Oymak, Tathagata Srimani, and Kerem Y. Camsari. Stochastic sparse attention for memory-bound inference. In *International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*. PMLR, 2026.

Mike Lewis, Shruti Bhosale, Tim Dettmers, Naman Goyal, and Luke Zettlemoyer. BASE layers: Simplifying training of large, sparse models. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6265–6274. PMLR, 2021.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020.

Opher Lieber et al. Jamba: A hybrid Transformer-Mamba language model. arXiv preprint arXiv:2403.19887, 2024.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11525–11538. Curran Associates, Inc., 2020.

Enzhe Lu et al. MoBA: Mixture of block attention for long-context LLMs. In *Advances in Neural Information Processing Systems*, volume 38, pp. 17790–17815. Curran Associates, Inc., 2025. doi: 10.52202/085713-0602.

Carlo Lucibello and Marc Mézard. Exponential capacity of dense associative memories. *Physical Review Letters*, 132(7):077301, 2024. doi: 10.1103/PhysRevLett.132.077301.

Wolfgang Maass. On the computational power of winner-take-all. *Neural Computation*, 12(11):2519–2535, 2000. doi: 10.1162/089976600300014827.

Yu. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4): 824–836, 2020. doi: 10.1109/TPAMI.2018.2889473.

Yury Malkov, Alexander Ponomarenko, Andrey Logvinov, and Vladimir Krylov. Approximate nearest neighbor algorithm based on navigable small world graphs. *Information Systems*, 45:61–68, 2014. doi: 10.1016/j.is.2013.10.006.

Stéphane G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989. doi: 10.1109/34.192463.

Daniel McFadden. Conditional logit analysis of qualitative choice behavior. In Paul Zarembka (ed.), *Frontiers in Econometrics*, pp. 105–142. Academic Press, New York, 1974.

Gonzalo Mena, David Belanger, Scott Linderman, and Jasper Snoek. Learning latent permutations with Gumbel-Sinkhorn networks. In *International Conference on Learning Representations*, 2018.

MiniMax. MiniMax-01: Scaling foundation models with lightning attention. arXiv preprint arXiv:2501.08313, 2025.

L. Mirsky. A dual of Dilworth's decomposition theorem. *The American Mathematical Monthly*, 78(8):876–877, 1971. doi: 10.2307/2316481.

Andriy Mnih and Geoffrey E. Hinton. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2008.

Alexander Mordvintsev, Ettore Randazzo, Eyvind Niklasson, and Michael Levin. Growing neural cellular automata. *Distill*, 5(2), 2020. doi: 10.23915/distill.00023.

Frederic Morin and Yoshua Bengio. Hierarchical probabilistic neural network language model. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, volume R5 of *Proceedings of Machine Learning Research*, pp. 246–252. PMLR, 2005. Reissued by PMLR on 30 March 2021.

Stephen Mussmann, Daniel Levy, and Stefano Ermon. Fast amortized inference and learning in log-linear models with randomly perturbed nearest neighbor search. In *Conference on Uncertainty in Artificial Intelligence*, 2017.

Roger B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981. doi: 10.1287/moor.6.1.58.

Ken M. Nakanishi. Scalable-softmax is superior for attention. arXiv preprint arXiv:2501.19399, 2025.

Maximilian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007. doi: 10.1109/JPROC.2006.887293.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the Web. Technical Report 1999-66, Stanford InfoLab, November 1999. URL http://ilpubs.stanford.edu:8090/422/. Previous number SIDL-WP-1999-0120.

Kostas Pagiamtzis and Ali Sheikholeslami. Content-addressable memory (CAM) circuits and architectures: A tutorial and survey. *IEEE Journal of Solid-State Circuits*, 41(3):712–727, 2006. doi: 10.1109/JSSC.2005.864128.

Inyoung Paik, Taeyeong Kwak, and Injung Kim. Capsule networks need an improved routing algorithm. In *Proceedings of The Eleventh Asian Conference on Machine Learning*, volume 101 of *Proceedings of Machine Learning Research*, pp. 489–502. PMLR, 2019.

Grigorios A. Pavliotis. *Stochastic Processes and Applications: Diffusion Processes, the Fokker–Planck and Langevin Equations*, volume 60 of *Texts in Applied Mathematics*. Springer, New York, 2014. doi: 10.1007/978-1-4939-1323-7.

Bo Peng et al. RWKV: Reinventing RNNs for the Transformer era. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 14048–14077. Association for Computational Linguistics, 2023.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. In *International Conference on Learning Representations*, 2024.

Hao Peng, Nikolaos Pappas, Dani Yogatama, Roy Schwartz, Noah A. Smith, and Lingpeng Kong. Random feature attention. In *International Conference on Learning Representations*, 2021.

Felix Petersen, Christian Borgelt, Hilde Kuehne, and Oliver Deussen. Differentiable sorting networks for scalable sorting and ranking supervision. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8546–8555. PMLR, 2021.

Gabriel Peyré and Marco Cuturi. Computational optimal transport with applications to data sciences. *Foundations and Trends in Machine Learning*, 11(5–6):355–607, 2019. doi: 10.1561/2200000073.

Tony A. Plate. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6(3):623–641, 1995. doi: 10.1109/72.377968.

Jon Postel. Internet Protocol. RFC 791 (STD 5), RFC Editor, https://www.rfc-editor.org/rfc/rfc791, September 1981.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=R8sQPpGCv0.

M. Prezioso, F. Merrikh-Bayat, B. D. Hoskins, G. C. Adam, K. K. Likharev, and D. B. Strukov. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature*, 521(7550):61–64, 2015. doi: 10.1038/nature14441.

Qwen Team. Qwen3-Next: Towards ultimate training & inference efficiency. Blog post, https://qwen.ai/blog?id=4074cca80393150c248e508aa62983f9cb7d27cd, September 2025. Model card: https://huggingface.co/Qwen/Qwen3-Next-80B-A3B-Instruct.

Jason Ramapuram, Federico Danieli, Eeshan Gunesh Dhekane, Floris Weers, Dan Busbridge, Pierre Ablin, Tatiana Likhomanenko, Jagrit Digani, Zijin Gu, Amitis Shidani, and Russell Webb. Theory, analysis, and best practices for sigmoid self-attention. In *International Conference on Learning Representations*, 2025.

Hubert Ramsauer et al. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-Depths: Dynamically allocating compute in transformer-based language models. arXiv preprint arXiv:2404.02258, 2024.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing Transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68, 2021. doi: 10.1162/tacl_a_00353.

Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Shimon Sakaguchi, Tomoyuki Yamaguchi, Takashi Nomura, and Masahiro Ono. Regulatory T cells and immune tolerance. *Cell*, 133(5):775–787, 2008. doi: 10.1016/j.cell.2008.05.009.

Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. Representation tradeoffs for hyperbolic embeddings. In *International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4460–4469. PMLR, 2018.

Michael E. Sander, Pierre Ablin, Mathieu Blondel, and Gabriel Peyré. Sinkformers: Transformers with doubly stochastic attention. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 3515–3530. PMLR, 2022.

Rik Sarkar. Low distortion Delaunay embedding of trees in hyperbolic plane. In Marc van Kreveld and Bettina Speckmann (eds.), *Graph Drawing: 19th International Symposium, GD 2011, Eindhoven, The Netherlands, September 21–23, 2011, Revised Selected Papers*, volume 7034 of *Lecture Notes in Computer Science*, pp. 355–366. Springer, 2011. doi: 10.1007/978-3-642-25878-7_34.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4938–4947, 2020.

Benjamin Scellier and Yoshua Bengio. Equilibrium Propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in Computational Neuroscience*, 11:24, 2017. doi: 10.3389/fncom.2017.00024.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear Transformers are secretly fast weight programmers. In *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9355–9366. PMLR, 2021.

J. Schnakenberg. Network theory of microscopic and macroscopic behavior of master equation systems. *Reviews of Modern Physics*, 48(4):571–585, 1976. doi: 10.1103/RevModPhys.48.571.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17456–17472. Curran Associates, Inc., 2022. doi: 10.52202/068431-1269.

Claude E. Shannon. Probability of error for optimal codes in a Gaussian channel. *Bell System Technical Journal*, 38(3):611–656, 1959.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (Learn at test time): RNNs with expressive hidden states. In *International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 57503–57522. PMLR, 2025.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to Transformer for large language models. arXiv preprint arXiv:2307.08621, 2023.

Zhong Sun, Giacomo Pedretti, Elia Ambrosi, Alessandro Bricalli, Wei Wang, and Daniele Ielmini. Solving matrix equations in one step with cross-point resistive arrays. *Proceedings of the National Academy of Sciences*, 116(10):4123–4128, 2019. doi: 10.1073/pnas.1815682116.

Dan Svenstrup, Jonas Meinertz Hansen, and Ole Winther. Hash embeddings for efficient word representations. In *Advances in Neural Information Processing Systems*, volume 30, pp. 4928–4936. Curran Associates, Inc., 2017.

Shawn Tan, Songlin Yang, Aaron Courville, Rameswar Panda, and Yikang Shen. Scaling stick-breaking attention: An efficient implementation and in-depth study. In *International Conference on Learning Representations*, 2025.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. Sparse Sinkhorn attention. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9438–9447. PMLR, 2020.

Paul Turán. On an extremal problem in graph theory. *Matematikai és Fizikai Lapok*, 48:436–452, 1941. In Hungarian.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6306–6315. Curran Associates, Inc., 2017.

A. VanderLugt. Signal detection by complex spatial filtering. *IEEE Transactions on Information Theory*, 10 (2):139–145, 1964. doi: 10.1109/TIT.1964.1053650.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Petar Veličković, Christos Perivolaropoulos, Federico Barbero, and Razvan Pascanu. Softmax is not enough (for sharp size generalisation). In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *PMLR*, 2025.

Gabriel D. Victora and Michel C. Nussenzweig. Germinal centers. *Annual Review of Immunology*, 40(1): 413–442, 2022. doi: 10.1146/annurev-immunol-120419-022408.

Roger Waleffe et al. An empirical study of Mamba-based language models. arXiv preprint arXiv:2406.07887, 2024.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics*, 38(5):146:1–146:12, 2019. doi: 10.1145/3326362.

C. S. Weaver and J. W. Goodman. A technique for optically convolving two functions. *Applied Optics*, 5(7): 1248–1249, 1966. doi: 10.1364/AO.5.001248.

Bernard Widrow and Marcian E. Hoff, Jr. Adaptive switching circuits. In *IRE WESCON Convention Record, Part 4*, pp. 96–104, August 1960.

Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing Transformers. In *International Conference on Learning Representations*, 2022.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024.

Liu Yang, Kangwook Lee, Robert Nowak, and Dimitris Papailiopoulos. Looped transformers are better at learning learning algorithms. In *International Conference on Learning Representations*, 2024a.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention Transformers with hardware-efficient training. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 56501–56523. PMLR, 2024b.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear Transformers with the delta rule over sequence length. In *Advances in Neural Information Processing Systems*, volume 37, pp. 115491–115522. Curran Associates, Inc., 2024c.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving Mamba2 with delta rule. In *International Conference on Learning Representations*, 2025.

Jingyang Yuan et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23078–23097. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1126.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. SoundStream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30: 495–507, 2022. doi: 10.1109/TASLP.2021.3129994.

Zihao Zeng, Yibo Miao, Hongcheng Gao, Hao Zhang, and Zhijie Deng. AdaMoE: Token-adaptive routing with null experts for mixture-of-experts language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 6223–6235. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.361.

Meiran Zhao, Bin Gao, Jianshi Tang, He Qian, and Huaqiang Wu. Reliability of analog resistive switching memory for neuromorphic computing. *Applied Physics Reviews*, 7(1):011301, 2020. doi: 10.1063/1.5124915.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, volume 35, pp. 7103–7114. Curran Associates, Inc., 2022. doi: 10.52202/068431-0515.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. ST-MoE: Designing stable and transferable sparse expert models. arXiv preprint arXiv:2202.08906, 2022.

A Proofs

A.1 Evidence

Proof of Proposition 1. Part (a). With an optional null weight $z \geq 0$, $w_t \geq \frac{1}{2}$ reads $e^{\tau s_t} \geq z + \sum_{j \neq t} e^{\tau s_j} \geq e^{\tau s_j}$ for every $j \neq t$, and $\tau > 0$ gives $s_t \geq s_j$.

Part (b). Under the factorization the posterior probability that the target is j, given the scores s, is $\ell(s_j)/\sum_k \ell(s_k)$ wherever the denominator is positive, because $g(s)$ and the uniform prior cancel. A randomized rule that names j with probability $q_j(s)$ succeeds, given s, with probability $\sum_j q_j(s) \Pr(\text{target} = j \mid s) \leq \max_j \Pr(\text{target} = j \mid s)$. Since ℓ is nondecreasing, a largest score attains that maximum. Averaging over s shows that no rule beats the argmax, and ties can be broken among the posterior maximizers without changing the value. For independent backgrounds with density f_0 and an independent target with density f_1, $p_t(s) = \prod_k f_0(s_k) \cdot f_1(s_t)/f_0(s_t)$, which is the factorization with $\ell = f_1/f_0$.

Part (c). Condition on the target's score. The target is the strict argmax when all $N-1$ independent background scores fall below c, which has probability $F(c)^{N-1}$; ties have probability zero. Solving $F(c)^{N-1} = \frac{1}{2}$ gives N_{ev}. □

Remark 8 (Exchangeability is not enough). Put the target uniformly at one of three positions. Give the other two the same score B, uniform on $(0,1)$, and the target an independent score with density $2x$ on $(0,1)$. The backgrounds are exchangeable and the marginal likelihood ratio $2x$ is increasing, yet the unique score that differs from the other two identifies the target surely, while the argmax succeeds only with probability $\Pr(T > B) = \mathbb{E}\, T = \frac{2}{3}$. The factorization in (b) fails because the background scores are not independent.

For random keys, $F = F_D$ is the distribution function of the cosine between a fixed unit vector and a uniform one in $\mathbb{R}^D$ ($D \geq 2$). Its density is $(1-x^2)^{(D-3)/2}/B(\frac{1}{2}, \frac{D-1}{2})$ on $(-1,1)$, and the tail expansion of Proposition 14(b) gives the rate $C(c)$ used in Section 3.

A.2 Gain

Proof of Proposition 2. Part (a). The largest-score sample is the target exactly when no sample lands above the target and at least one lands on it. No sample lands above it with probability $(1-q)^M$, and none lands above it or on it with probability $(1-q-w)^M$; the difference is the success probability. With $q = 0$, $1-(1-w)^M \geq \frac{1}{2}$ is $w \geq 1 - 2^{-1/M}$.

Part (b). The number of target samples is $\text{Bin}(M, w)$. Its probability of exceeding $M/2$ increases strictly with w on $(0,1)$, by a coupling through uniform random variables. At $w = \frac{1}{2}$, symmetry gives probability $\frac{1}{2}$ for odd M and $\frac{1}{2}\big(1 - \Pr\{\text{Bin}(M, \frac{1}{2}) = M/2\}\big) < \frac{1}{2}$ for even M, so the half-success threshold is $\frac{1}{2}$ for odd M and larger for even M.

Part (c). Along a path from the root to leaf j the factors $Z_{\text{child}}/Z_{\text{parent}}$ telescope to $e^{\tau s_j}/Z_{\text{root}}$, the flat softmax. For a product score $a_i + b_j$ the partition function factors, so the target's weight is the product of its two half-weights. Each half has $k-1$ other shortlisted entries with half-scores at most u, so its half-weight is at least $[1 + (k-1)e^{-\tau(c/2-u)}]^{-1}$, with equality when the others all score u. The product of the two bounds is at least $\frac{1}{2}$ exactly when $(k-1)e^{-\tau(c/2-u)} \leq \sqrt{2} - 1$, that is $\tau(c-2u) \geq 2\ln((1+\sqrt{2})(k-1))$. For $k = 1$, inclusion alone gives the target all the mass. □

Remark 9 (Soft at every level is not enough). Take two groups at the root, one holding two leaves and the other one leaf. Give the groups equal parent probabilities and every leaf the score zero. The leaves get probabilities $(\frac{1}{4}, \frac{1}{4}, \frac{1}{2})$, while the flat softmax gives each $\frac{1}{3}$. Only parent weights proportional to the subtree partition functions reproduce the flat softmax.

The mean-field forms. Replacing the background sum by $Nm(\tau)$ gives the target weight $e^{\tau c}/(e^{\tau c} + Nm(\tau))$, whose majority onset is $N^* = e^{\tau c}/m(\tau)$. With $q = 0$ the select onset is $N_M^* = N^*(1/w_M - 1)$, and since $w_M \sim \ln 2/M$, the log of the total count $1 + N_M^*$ gains one nat per nat of $\ln M$. These follow by solving for N. For independent backgrounds with $m(2\tau) < \infty$, Chebyshev's inequality bounds the relative error of the replacement: $\Pr\big(|\sum_j e^{\tau z_j}/(Nm(\tau)) - 1| > u\big) \leq (m(2\tau)/m(\tau)^2 - 1)/(Nu^2)$. Where this bound is large,

the mean-field onset is not certified. For $z \sim \mathcal{N}(0, 1/D)$, completing the square gives the mean-field onset $\ln N^* = \tau c - \tau^2/(2D)$, which is largest at $\tau = cD$. In the large-D limit of the Gaussian surrogate it is the true onset only for $\tau < cD$. Beyond $\tau = cD$ the background sum at that onset freezes onto its largest terms, and the true onset keeps rising toward the evidence wall instead of falling (Lucibello & Mézard, 2024).

A.3 Abstention

Proof of Proposition 3. Part (a). Dividing numerator and denominator by e^b gives $a = 1/(1 + Be^{-b})$, which is at least $\frac{1}{2}$ exactly when $e^b \geq B$. With $B = Nm(\tau)$ this is $\ln N \leq b - \ln m(\tau)$. Under $b = b_0 + \ln N$ the mean-field abstention is $e^{b_0}N/(e^{b_0}N + Nm(\tau)) = e^{b_0}/(e^{b_0} + m(\tau))$. On a present query the target's weight is $e^{\tau c}/(e^{\tau c} + Ne^{b_0} + Nm(\tau))$, which falls below $\frac{1}{2}$ once $\ln N > \tau c - \ln(e^{b_0} + m(\tau))$.

Part (b). The rule abstains when all N independent scores fall below θ, with probability $F(\theta)^N$. Solving $F(\theta)^N = \frac{1}{2}$ gives the onset. If $F(\theta) = 1$ it never fails, and if $F(\theta) < 1$ it fails eventually, although no normalizer is involved.

Part (c). Given the row, the M samples are independent, and each lands above θ with probability p_θ. □

A finite number of competitive walkers turns (a)'s decision into a binomial absorption event. It is not deterministically equivalent to the half-mass decision, but converges to it, away from equality, as the number of samples grows.

A.4 Tethers

Proof of Proposition 4. The fixed point. A row-stochastic P satisfies $\|PX\|_\infty \leq \|X\|_\infty$ in the largest-row norm, so the map $X \mapsto (1-\alpha)X_0 + \alpha PX$ contracts by $\alpha < 1$. Its unique fixed point is the Neumann series $(1-\alpha)\sum_{k\geq 0} \alpha^k P^k X_0 = RX_0$, and every row of R sums to one.

Part (a). The cosines of output row i with the unit inputs are the entries of $(RG)_{i\cdot}$, divided by the row's norm. Normalizing multiplies them all by the same positive number, and a strict winner implies a nonzero row. For orthonormal rows $G = I$.

Part (b). On a cycle of length m, the resolvent puts weight $(1-\alpha)\alpha^r/(1-\alpha^m)$ on the token at cyclic distance $r = 0, \ldots, m-1$, the geometric series over returns. The own weight, $r = 0$, is the largest for every $\alpha < 1$. A token d steps before the cycle has own weight $1-\alpha$, weight $(1-\alpha)\alpha^r$ on the path token $r < d$ steps ahead, and weight $(1-\alpha)\alpha^{d+r}/(1-\alpha^m)$ on the cycle token at distance r. The own weight beats every path token's, since $\alpha^r < 1$. Among the cycle tokens the entry point's weight, $(1-\alpha)\alpha^d/(1-\alpha^m)$, is the largest, and the own weight beats it exactly when $1 - \alpha^m > \alpha^d$.

Part (c). Starting from the token's own input, write d_t for its own coefficient minus its partner's. Then $d_0 = 1$ and $d_{t+1} = 1 - \alpha - \alpha d_t$, so $d_t = [1 - \alpha + 2\alpha(-\alpha)^t]/(1+\alpha)$, which is positive at every even t and, at odd t, exactly when $1 - \alpha > 2\alpha^{t+1}$. □

Near-orthogonal inputs. With coherence $\eta = \max_{i\neq j} |G_{ij}| < 1$, the difference of the two correlations in (a) is $(R_{ii} - R_{ij})(1 - G_{ij}) + \sum_{k\neq i,j} R_{ik}(G_{ki} - G_{kj})$. Since the row sums to one, it is at least $(R_{ii} - R_{ij})(1 - \eta) - 2\eta(1 - R_{ii} - R_{ij})$. So $R_{ii} > R_{ij}$ together with $(R_{ii} - R_{ij})(1-\eta) > 2\eta(1 - R_{ii} - R_{ij})$ for every $j \neq i$ is sufficient. Identical inputs cannot be decoded uniquely whatever R is.

Two-cycles. For a symmetric, strict, self-excluding nearest-neighbour score, a directed cycle $i_1 \to i_2 \to \cdots \to i_m \to i_1$ with $m \geq 3$ would need $s(i_1, i_2) < s(i_2, i_3) < \cdots < s(i_m, i_1) < s(i_1, i_2)$, because each token points to its strictly best neighbour and the score is symmetric. So every cycle has length two.

A.5 Gain control

Proof of Proposition 5. Part (a). For a probability vector p the softmax Jacobian is $J = \mathrm{diag}(p) - pp^\top$. For $\|v\|_\infty \leq 1$, $\|Jv\|_1 = \mathbb{E}_p|v - \mathbb{E}_p v| \leq \sqrt{\mathrm{Var}_p(v)} \leq 1$, so softmax is 1-Lipschitz from ℓ_∞ to ℓ_1; two equal probabilities and $v = (1, -1)$ attain the bound. On the row ball, Cauchy–Schwarz gives $\max_{i,j} |\tau x_i^\top x_j -$

$\tau y_i^\top y_j| \le 2\tau R\|X-Y\|_{\mathrm{r}}$. Splitting $P(X)X - P(Y)Y = P(X)(X-Y) + (P(X)-P(Y))Y$ bounds the first term by $\|X-Y\|_{\mathrm{r}}$ and the second by $2\tau R^2\|X-Y\|_{\mathrm{r}}$, so $\mathrm{Lip}(F) \le \alpha(1+2\tau R^2)$. Each output row is βx_{0i} plus α times a convex combination of rows of norm at most R, so the ball is invariant when $R \ge R_0$. A contraction of a closed set into itself has a unique fixed point that attracts every start, by the summable geometric series of consecutive differences.

Part (b). On the ball of radius R_0 every logit difference in a row is at most $2\tau R_0^2$, so $P_{ij}(X) \ge e^{-2\tau R_0^2}/m_i \ge \ell_i$ for $j \in V_i$. The image of the ball lies in $\mathcal{K}_1 = \prod_i [\beta x_{0i} + \alpha \overline{B}(0, R_0)]$. For $X \in \mathcal{K}_1$ write $x_j = \beta x_{0j} + \alpha z_j$ with $\|z_j\| \le R_0$, and split each row's probabilities into the common ℓ_i and the remainder of mass $1 - m_i\ell_i$:

$$F_i(X) = c_i + \alpha\beta \sum_{j \in V_i} (P_{ij} - \ell_i) x_{0j} + \alpha^2 \sum_{j \in V_i} P_{ij} z_j.$$

The last two terms have norm at most $\alpha\beta(1-m_i\ell_i)R_0 + \alpha^2 R_0 = r_i$, so $F(\mathcal{K}_1) \subseteq \mathcal{K}$. Also $\|c_i - \beta x_{0i}\| + r_i \le \alpha R_0$, so $\mathcal{K} \subseteq \mathcal{K}_1$. This gives invariance and absorption in two steps, and every fixed point in the ball lies in $\mathcal{K}$. Part (a)'s argument on the convex set $\mathcal{K}$, whose rows have norm at most R_{eff}, then gives the contraction, and absorption gives attraction from the whole ball.

Part (c). For X in the candidate ball, $\|F(X) - \widehat{X}\|_{\mathrm{r}} \le qr + e \le r$, so the ball is invariant and part (a)'s contraction argument applies; the fixed point X_* satisfies $\|X_* - \widehat{X}\|_{\mathrm{r}} \le q\|X_* - \widehat{X}\|_{\mathrm{r}} + e$. For the refinement, a zero-sum signed vector a has $\|\sum_j a_j y_j\| \le (D/2)\sum_j |a_j|$, since its positive and negative parts are equal masses times two convex combinations of rows, at distance at most D. Applying this to a row of $P(X) - P(Y)$, whose ℓ_1 norm is at most $2\tau R\|X-Y\|_{\mathrm{r}}$, replaces $2\tau R^2$ by $\tau R D$. □

A.6 Transport cannot write

Proposition 10 (Transport cannot write)**.** *Let $X_{t+1} = P_t(X_t)X_t$ with every P_t row-stochastic and nonnegative, chosen adaptively from the current state. Every output row lies in the convex hull of the rows of X_0, and* $\mathrm{rank}\, X_t \le \mathrm{rank}\, X_0$. *A common pointwise nonlinear writer can raise the rank and leave the hull.*

Proof. Given the realized input, the product forms convex combinations of its rows, and its rank is at most that of its right factor; iterating gives both statements. For the writer, the rows $(1, -1)$ and $(-1, 1)$ form a matrix of rank one, and the coordinatewise ReLU maps it to the identity, of rank two. □

The statement concerns one shared left transport. Several heads with different channel maps break its premise: from the rank-one rows $(1, 0)$ and $(0, 0)$, an identity head keeping the first channel and a swap head writing it to the second sum to the rank-two identity. A shared linear channel map alone still cannot raise the rank. None of this sets the rank a task requires, or gives an accuracy bound without a specified readout.

A.7 Address and cargo

Proof of Theorem 6. Condition on $G = g$ and on the target's class y. The target's unhalved score is $c + g_y$. Each other item's unhalved score is $Z_j + g_{Y_j}$ with an independent uniform class, so it has distribution function F_g, independently across items given g. All $N-1$ fall below $c + g_y$ with probability $F_g(c+g_y)^{N-1}$, and ties have probability zero. Averaging over the target's class and over G gives p_N. With no cargo every threshold is c. For unequal class probabilities the same steps give weighted sums. □

The curve decreases in N term by term, strictly where $0 < F_G(c + G_y) < 1$ with positive probability. Its half-success count is where it crosses $\frac{1}{2}$. The evaluation quoted in Section 7 integrates the exact conditional law with six independently scrambled Sobol rules (2^{15} points each); their spread is a diagnostic, not an error bar. Evaluated on the archived coarse grid, the same model gives a tied onset within 0.002 of the archive's. The inputs, seeds and hashes are in `data/separation_quadrature/`.

Proof of Proposition 7. Each coordinate maximizes its own addend independently, so the global argmaxes are exactly the pairs of half-argmaxes, and every Cartesian top-k list, which keeps a top half-score on each

side, contains one of them. A selection that is a function of the clean scores alone cannot change when the cargo does. Comparing each perturbed pair score with the winner's gives both conditions. □

Inclusion for random codebooks. Condition on the query halves, the target codewords and the target's half-scores t_A, t_B. Put $\rho_h = \|q_h\|_2/\sqrt{2}$, and let F_{64} be the distribution function of a spherical cosine in $\mathbb{R}^{64}$. The other $B-1$ codewords in half h score independently with distribution function $F_{64}(x/\rho_h)$, so the target is included in a top-k product shortlist with conditional probability $\prod_{h\in\{A,B\}} \Pr\{\mathrm{Bin}(B-1, 1-F_{64}(t_h/\rho_h)) \le k-1\}$, by counting the codewords above the target. Averaging over the query and the targets gives the unconditional probability. For imperfect queries the half-norms fluctuate, so setting $\rho_h = \frac{1}{2}$ throughout is an approximation, not the source's distribution.

A.8 The price of order

Fix a query. Key $j \in \{1, \dots, n\}$ receives the logit $\tau c_j + b_j$, where c_j is its content score and b_j is a positional term: any function of the key's position for this query, such as a recency bias, a linear distance penalty or the positional part of a learned score. Let $B = \max_j b_j - \min_j b_j$ be its range. Two identical writes ($c_j = c_k$) at positions $j < k$ are *ordered with ratio* $r > 1$ when the later one receives at least r times the weight of the earlier one. The two weights share one normalizer, so this holds if and only if $b_k - b_j \ge \ln r$, whatever the other keys are. Write $t(n, h)$ for the number of edges of the Turán graph (Turán, 1941), the complete h-partite graph on n vertices with parts as equal as possible.

Proposition 11 (The price of order)**.** *Let* $K = \lfloor B/\ln r \rfloor$.

(a) At most $t(n, K+1)$ *pairs are ordered with ratio* r*. A staircase attains this: split the positions into* $K+1$ *consecutive blocks, as equal in size as possible, and set* $b_j = k \ln r$ *on block* k*. As* $n \to \infty$ *the ordered fraction is at most* $K/(K+1)$*, so ordering a fraction* ϕ *of the pairs needs* $B \ge \lceil \phi/(1-\phi) \rceil \ln r$*.*

(b) A linear term $b = \lambda s$*, with positions* s *spread uniformly over* $[0, 1]$*, orders a fraction* $(1 - \ln r/\lambda)_+^2$ *of the pairs in the continuum limit. At a large range* $B = \lambda$ *it leaves about* $2 \ln r/B$ *of the pairs unordered, twice the staircase's* $\ln r/B$*.*

(c) (Mean field.) Let the N *background scores be exchangeable with* $m(\tau) = \mathbb{E}e^{\tau c}$*, let the target's score be* c_t*, and require every position to stay retrievable: a single target at the position with the smallest positional term keeps at least half the mass. This costs* $\frac{1}{N}\sum_j e^{b_j - \min b} \le e^{\Delta}$*, where* $\Delta = \tau c_t - \ln m(\tau) - \ln N$ *is the unbiased router's margin above its gain wall. Over all positional terms that meet it, the largest ordered fraction tends, as* $N \to \infty$ *with* Δ *held fixed, to*

$$F^*(\Delta) = \max\Big\{1 - \sum_{k\ge 0} f_k^2 \;:\; f_k \ge 0,\ \sum_k f_k = 1,\ \sum_k f_k r^k \le e^{\Delta}\Big\},$$

attained by a staircase whose block k *holds a fraction* f_k *of the positions. The maximizer has the form* $f_k = (\mu - \nu r^k)_+$ *with* $\nu \ge 0$*. At the gain wall (*$\Delta = 0$*) nothing can be ordered, and* $F^*(\Delta) < 1$ *for every finite* Δ*.*

(d) A hard machine that compares timestamps, or overwrites on write, orders every pair and spends none of this margin.

Proof. A partial order. For positions $j < k$ write $j \prec k$ when $b_k - b_j \ge \ln r$, so that the ordered pairs are exactly the pairs related by $\prec$. The relation is transitive, since $j \prec k \prec l$ gives $b_l - b_j \ge 2\ln r$, so it is a strict partial order. A chain $j_0 \prec j_1 \prec \cdots \prec j_m$ has $b_{j_m} - b_{j_0} \ge m \ln r$, so $m \le B/\ln r$ and every chain has at most $K+1$ elements.

Levels. Let $\ell(j) + 1$ be the number of elements of the longest chain that ends at j, so that $\ell(j) \in \{0, \dots, K\}$. If $j \prec k$, appending k to a longest chain ending at j gives $\ell(k) \ge \ell(j) + 1$. Two positions on the same level are therefore never ordered with each other: the levels are Mirsky's partition (Mirsky, 1971) into antichains. Along a longest chain ending at j the positional term rises by at least $\ln r$ per step, so $b_j - \min b \ge \ell(j) \ln r$.

Table 5: The price of order in the constructed arrow test ($N = 1024$, $D = 128$, $r = 9$). Δ is the unbiased router's margin above its gain wall. $F^*(\Delta)$ is the largest fraction of duplicate pairs that any positional term can order while every position stays retrievable (mean field, Proposition 11(c)); steps is the number of steps of the staircase that attains it. The staircase designed for $\Delta - \frac{1}{2}$ was run on the test's substrate (20,000 trials): latest write is the fraction of duplicate pairs in which the later copy holds at least half the mass and at least r times the earlier copy's weight, and lowest step is the fraction of single targets on its lowest step that hold at least half the mass. The best linear term is the largest slope λ within the same budget, and ordered is the fraction of pairs it orders, $(1 - \ln r/\lambda)^2$.

		any positional term		staircase for $\Delta - \frac{1}{2}$			best linear term	
τ	Δ (nats)	$F^*(\Delta)$	steps	$F^*(\Delta - \frac{1}{2})$	latest write	lowest step	λ	ordered
10	2.68	0.60	3	0.55	0.55	1.00	4.1	0.22
20	11.52	0.86	8	0.85	0.85	0.96	14.2	0.71

Replacing b by $b'_j = \ell(j) \ln r$ therefore keeps every ordered pair, because $\ell(k) - \ell(j) \geq 1$ whenever $j \prec k$, and lowers every positional term relative to its minimum.

Part (a). Ordered pairs join different levels. If n_k positions sit on level k, there are at most $\frac{1}{2}\big(n^2 - \sum_k n_k^2\big)$ of them, the number of pairs split by a partition into at most $K+1$ parts. Over integer parts this is largest when the parts are as equal as possible, where it equals $t(n, K+1)$. By Cauchy–Schwarz, $\sum_k n_k^2 \geq n^2/(K+1)$, so $t(n, K+1) \leq \frac{K}{K+1}\frac{n^2}{2}$ and $t(n, K+1)/\binom{n}{2} \to K/(K+1)$. The staircase has range $K \ln r \leq B$. Positions in different blocks differ by at least $\ln r$ in the right direction and positions in the same block not at all, so it orders exactly the $t(n, K+1)$ cross-block pairs. Ordering a fraction ϕ in the limit requires $K/(K+1) \geq \phi$, that is $\lfloor B/\ln r \rfloor \geq \phi/(1-\phi)$.

Part (b). Two independent uniform positions are ordered when they differ by at least $x = \ln r/\lambda$, which has probability $(1-x)_+^2$. The unordered fraction $1 - (1-x)^2 = 2x - x^2$ is about $2 \ln r/B$ at large B, while the staircase leaves $1/(K+1) \approx \ln r/B$.

Part (c). A single target at the position j_0 with the smallest positional term has weight $e^{\tau c_t + b_{j_0}}$ against the background sum $\sum_j e^{\tau c_j + b_j}$, which is $m(\tau) \sum_j e^{b_j}$ in mean field. Keeping half the mass is therefore $\sum_j e^{b_j - b_{j_0}} \leq e^{\tau c_t - \ln m(\tau)} = N e^{\Delta}$. The constraint only gets easier when positional terms fall relative to their minimum, so by the level argument the optimum is attained by a level assignment $b_j = \ell_j \ln r$ with $\sum_j r^{\ell_j} \leq N e^{\Delta}$. For given level counts n_k, sorting the levels by position orders every cross-level pair, and no arrangement orders a same-level pair, so the count is $\frac{1}{2}(N^2 - \sum_k n_k^2)$. With $f_k = n_k/N$ the ordered fraction tends to $1 - \sum_k f_k^2$, and the constraint reads $\sum_k f_k r^k \leq e^{\Delta}$. This is a strictly convex quadratic program, and its KKT conditions give $f_k = (\mu - \nu r^k)_+$ with $\nu \geq 0$. If $\Delta = 0$, then $\sum_k f_k r^k \leq 1 = \sum_k f_k$ with $r^k > 1$ for $k \geq 1$ forces $f_0 = 1$ and $F^* = 0$. For finite Δ, the levels above k_0 carry mass at most $e^{\Delta} r^{-k_0}$. Choosing k_0 with $e^{\Delta} r^{-k_0} \leq \frac{1}{2}$ leaves mass at least $\frac{1}{2}$ on $k_0 + 1$ levels, so $\sum_k f_k^2 \geq \frac{1}{4}/(k_0 + 1) > 0$ and $F^* < 1$.

Part (d). A timestamp comparison selects the later write, and overwrite leaves only one. Neither adds a positional term to the scores, so nothing is taken from the retrieval margin. □

Remark 12 (The constructed arrow test). The constructed arrow test has $N = 1024$ keys in $D = 128$ dimensions, exact targets $c_t = 1$, background cosines of uniform random unit vectors, and $r = 9$: the later copy must hold 90% of the pair's weight. There, $\Delta = 2.68$ nats at $\tau = 10$ and 11.52 at $\tau = 20$, and Table 5 gives $F^* = 0.60$ and 0.86. No positional term of any shape reaches 0.9 latest-write success while every position stays retrievable. The optimal staircase orders nearly three times as many pairs as the best linear term at $\tau = 10$. A staircase designed for $\Delta - \frac{1}{2}$ reaches its mean-field value in simulation, with a single target on its lowest step still retrieved in 95.9% of trials, so the bound is attained beyond mean field.

A.9 Entropy production

Let P be a finite Markov kernel with a strictly positive stationary law π, and split the stationary flow $Q_{ij} = \pi_i P_{ij}$ as $Q = S + A$ with $S^\top = S$ and $A^\top = -A$. In the notation of Bruce (2026a, §3.2), $S_{ij} = \pi_i \bar{P}_{ij}$ and $A_{ij} = \pi_i J_{ij}$, where $\bar{P}$ and J come from the π-weighted reversal. Assume that every edge with positive flow also has positive reverse flow.

Theorem 13 (Entropy production at second order)**.** *With sums over ordered pairs, define the entropy production per step as* $\sigma = \sum_{i,j} Q_{ij} \ln(Q_{ij}/Q_{ji}) = \frac{1}{2}\sum_{i,j}(Q_{ij} - Q_{ji}) \ln(Q_{ij}/Q_{ji})$*. If* $|A_{ij}| \le r S_{ij}$ *with* $r < 1$ *on every edge with positive flow, then*

$$\sigma = 2 \sum_{i,j} \frac{A_{ij}^2}{S_{ij}} + 2 \sum_{k \ge 1} \frac{1}{2k+1} \sum_{i,j} \frac{A_{ij}^{2k+2}}{S_{ij}^{2k+1}},$$

and the nonnegative remainder after the first term is at most $\frac{2}{3(1-r^2)} \sum_{i,j} A_{ij}^4 / S_{ij}^3$*. In particular* σ *is even under reversing the current at fixed* S*.*

Proof. Pairing (i, j) with (j, i) gives the second form of σ, which equals $\sum_{i,j} A_{ij} \ln[(S_{ij} + A_{ij})/(S_{ij} - A_{ij})]$. For $|x| < 1$, integrating the geometric series of $2/(1 - x^2)$ gives $\ln[(1 + x)/(1 - x)] = 2\sum_{k \ge 0} x^{2k+1}/(2k + 1)$. Substituting $x = A_{ij}/S_{ij}$ gives the series, each term of which is nonnegative and even. For $k \ge 1$, $1/(2k+1) \le \frac{1}{3}$ and successive powers shrink by at most r^2, which bounds the remainder. Edges with zero flow contribute nothing. □

A one-way flow with zero reverse flow makes σ infinite in this convention, and the expansion does not apply. The physical reading assumes a thermodynamic model and its units; an attention matrix is not a measured energy dissipation.

A.10 Conservation of routing information

A store holds N items, item i with key k_i and value x_i. A query names one item, the target J, or none. All quantities are in nats, and $h(\varepsilon) = -\varepsilon \ln \varepsilon - (1 - \varepsilon) \ln(1 - \varepsilon)$ is the binary entropy. Each part prices one task within its own class of machines: what must be paid, and in which currency. The corollary shows that no fixed allotment of resources survives a growing N.

Proposition 14 (Conservation of routing information)**.** *(a)* Identification. *Let* J *be uniform on* $\{1, \dots, N\}$ *and let a rule, randomized or not, name* $\hat{J}$ *from an observation* Y*. If* $\Pr(\hat{J} \ne J) \le \varepsilon$*, then* $I(J; Y) \ge (1 - \varepsilon) \ln N - \ln 2$*.*

(b) The key channel. *Let the* $N - 1$ *background keys be independent and uniform on the unit sphere in* $\mathbb{R}^d$*, independent of the query* q*, and let the target's key have cosine* $c \in (0, 1)$ *with* q*. Consider readers that rank keys by their cosine with* q*: an argmax, a top-*k *stage, a gate on the largest score, or a soft router whose weights increase with the score. Such a reader's final selection is the target, or it puts a strict majority on the target, only if the target is the argmax. That has probability* $F_d(c)^{N-1}$*, where* F_d *is the distribution function of the cosine, and the probability is one half at* $N_{\mathrm{ev}} = 1 + \ln 2/(-\ln F_d(c))$*. With* $C(c) = -\frac{1}{2}\ln(1 - c^2) = \frac{1}{2}\ln(1 + \mathrm{SNR})$ *and* $\mathrm{SNR} = c^2/(1 - c^2)$*,*

$$-\ln\big(1 - F_d(c)\big) = (d - 1)\, C(c) + \ln\big(c\sqrt{2\pi(d - 1)}\big) + \frac{c^{-2} - \frac{3}{4}}{d - 1} + O\big(d^{-2}\big).$$

(c) Expression. *Let a soft router give the target the weight* $w_t = e^{s_t}/(z + \sum_{j=1}^{N} e^{s_j})$*, where* $z \ge 0$ *is an optional null weight and all* N *logits lie in a window of width* R*. If* $w_t \ge w^*$*, then* $N - 1 \le e^R(1 - w^*)/w^*$*. A majority (*$w^* = \frac{1}{2}$*) therefore needs* $\ln(N - 1) \le R$*. A reader of type (b) that selects pays nothing here.*

(d) Memory. *A one-pass machine reads the store into a state* M *that takes at most* e^S *values, possibly using randomness. The query arrives afterwards, and the answer is computed from* M *and the query*

alone. Let the keys be fixed and the values independent and uniform on a set of e^v elements. If each item's value is returned correctly with probability at least $1-\varepsilon$, where $\varepsilon \le 1-e^{-v}$ (no worse than guessing), then

$$S \;\ge\; N\big(v - h(\varepsilon) - \varepsilon \ln(e^v - 1)\big) \;\ge\; N\big((1-\varepsilon)v - \ln 2\big).$$

For membership, let the stored keys be an N-subset of a universe of U keys, let the machine be deterministic, and let it report every stored key present and at most a fraction $\varepsilon \in (0,1]$ of the absent keys. Then $S \ge \ln \binom{U}{N} - \ln \binom{\lfloor N+\varepsilon(U-N)\rfloor}{N}$, which tends to $N \ln(1/\varepsilon)$ as $U \to \infty$ at fixed N.

(e) Order. *A one-pass machine reads N distinct, known writes, arriving in a uniformly random order, into a state of at most e^S values. It then answers "which of i and j came first?" from the state alone. If all its answers are correct together with probability at least $1-\varepsilon$, then $S \ge (1-\varepsilon)\ln N! - \ln 2$. A soft router that carries order in its weights is priced by Proposition 11.*

Proof. Part (a). Fano's inequality (Cover & Thomas, 2006), with the actual error probability $p \le \varepsilon$, gives $H(J \mid Y) \le h(p) + p \ln(N-1) \le \ln 2 + \varepsilon \ln N$. So $I(J;Y) = \ln N - H(J \mid Y) \ge (1-\varepsilon)\ln N - \ln 2$.

Part (b). Suppose a background key has a cosine at least the target's. Then a selecting reader does not pick the target, and a router whose weights increase with the score gives that key at least the target's weight. The two weights sum to at most one, so the target's weight is at most one half. The background cosines are independent with distribution function F_d and the target's cosine is c, so the target is the strict argmax with probability $F_d(c)^{N-1}$. This is one half exactly at N_{ev}.

For the tail, a coordinate of a uniform unit vector in $\mathbb{R}^d$ has density $(1-x^2)^m / B(\frac12, m+1)$ with $m = (d-3)/2$. Put $x = c+t$, $a = 2c/(1-c^2)$ and $b = (1+c^2)/(1-c^2)^2$. Then $1-x^2 = (1-c^2)(1-at-t^2/(1-c^2))$ and $m\ln(1-at-t^2/(1-c^2)) = -m(at+bt^2+O(t^3))$, so

$$1 - F_d(c) = \frac{(1-c^2)^m}{B(\frac12, m+1)} \int_0^{1-c} e^{-m(at+bt^2+O(t^3))}\,dt = \frac{(1-c^2)^{m+1}}{2cm\,B(\frac12, m+1)}\Big(1 - \frac{1+c^2}{2c^2 m} + O(m^{-2})\Big)$$

by Watson's lemma ($\int_0^\infty e^{-m(at+bt^2)}dt = \frac{1}{am}(1 - \frac{2b}{a^2 m} + O(m^{-2}))$; the integral beyond a fixed neighbourhood of $t=0$ is exponentially smaller). Next, $1/B(\frac12, m+1) = \Gamma(m+\frac32)/(\Gamma(\frac12)\Gamma(m+1)) = \sqrt{(m+1)/\pi}\,(1 - \frac{1}{8(m+1)} + O(m^{-2}))$. Put $n = d-1$, so that $m+1 = n/2$ and $(1-c^2)^{m+1} = e^{-nC(c)}$, and take logarithms. The leading terms give $nC(c) + \ln(c\sqrt{2\pi n})$. The terms of order $1/n$ add up to $-2 + (1+c^2)/c^2 + \frac14 = c^{-2} - \frac34$: the first comes from $1/(2cm) = 1/(c(n-2))$, the second from Watson's correction and the third from the gamma ratio.

Part (c). Every other logit is at least $s_t - R$ and $z \ge 0$, so $w_t \le e^{s_t}/(e^{s_t} + (N-1)e^{s_t - R}) = 1/(1 + (N-1)e^{-R})$. Then $w_t \ge w^*$ gives $(N-1)e^{-R} \le (1-w^*)/w^*$. A selecting reader returns the argmax, whatever the gap.

Part (d). Write $X = (X_1, \dots, X_N)$ for the values and $\hat X_i$ for the answer to query i. Given M, the answer $\hat X_i$ depends on X only through M. So data processing and Fano's inequality give $H(X_i \mid M) \le H(X_i \mid \hat X_i) \le h(\varepsilon) + \varepsilon \ln(e^v - 1)$, since $h(p) + p\ln(e^v-1)$ increases in the error probability p up to the chance level $1 - e^{-v}$. Then $H(X \mid M) \le \sum_i H(X_i \mid M)$, and $S \ge H(M) \ge I(X;M) = Nv - H(X \mid M)$. The weaker form uses $h(\varepsilon) \le \ln 2$ and $\ln(e^v - 1) < v$. For membership, each state m accepts a set $A(m)$ of keys. A store K in state m has $K \subseteq A(m)$ and $|A(m) \setminus K| \le \varepsilon(U-N)$, so $|A(m)| \le N + \varepsilon(U-N)$. At most $\binom{\lfloor N+\varepsilon(U-N)\rfloor}{N}$ stores can therefore share a state, while all $\binom{U}{N}$ stores must be placed among the e^S states. For fixed N the ratio of the binomials tends to ε^{-N} as $U \to \infty$ (Carter et al., 1978).

Part (e). The answers to all pairs determine the order, so the machine recovers the uniformly random permutation π with probability at least $1-\varepsilon$. Fano's inequality gives $H(\pi \mid M) \le \ln 2 + \varepsilon \ln N!$, and so $S \ge H(M) \ge I(\pi; M) \ge (1-\varepsilon)\ln N! - \ln 2$. □

Corollary 15 (A wall for constant resources)**.** *Fix the dimension d and match cosine $c < 1$, the logit window R, the state size S and the positional range B. Each of the following holds once N is large enough.*

(i) A reader of type (b) succeeds with probability below one half (once $N > N_{\mathrm{ev}}$).

(ii) A soft router puts a majority on no target (once $N-1 > e^R$).

(iii) A one-pass machine cannot return an arbitrary stored value with error at most $\varepsilon < 1 - e^{-v}$ (once $N(v - h(\varepsilon) - \varepsilon \ln(e^v - 1)) > S$).

(iv) It cannot keep a recoverable order (once $(1-\varepsilon)\ln N! - \ln 2 > S$).

(v) A soft router cannot order a fraction $\phi > K/(K+1)$ of the pairs, with $K = \lfloor B/\ln r \rfloor$ (once $t(N, K+1) < \phi\binom{N}{2}$).

Each wall moves only when its own ledger is paid:

- *dimension or match quality for (i);*
- *logit range or a selecting read for (ii);*
- *state for (iii) and (iv);*
- *positional range for (v).*

Leaving the class also moves it: reading the question before the store, or reading the store twice, takes a machine outside (iii).

Proof. Parts (b) to (e) of Proposition 14 and Proposition 11(a); for (v), $t(N, K+1)/\binom{N}{2}$ tends to $K/(K+1) < \phi$. For $c < 1$, N_{ev} is finite. For $\varepsilon < 1 - e^{-v}$ the per-item term in (d) is positive, since $v - h(\varepsilon) - \varepsilon \ln(e^v - 1)$ is the divergence from the uniform law of the law that puts mass $1-\varepsilon$ on the true value and spreads the rest evenly. □

Remark 16 (Who pays which ledger).

- *Softmax attention* pays memory in full, with a cache that grows with N. It pays expression with its logit scale or a $\ln N$ schedule, and buys recency from the positional range of Proposition 11.
- *A recurrent machine with a fixed state* gets direction and recency by construction. At an error $\varepsilon < 1 - e^{-v}$ it cannot pay memory once N passes $S/(v - h(\varepsilon) - \varepsilon \ln(e^v - 1))$, unless the question comes first, which takes it outside Proposition 14(d): its state then needs to hold only the answer.
- *A max-statistic head* pays a constant $\ln(k-1)$ for expression, independent of N, and stops at the key channel.
- *Timestamps* pay for order with $N \ln N \geq \ln N!$ nats of storage. *Overwrite* answers only latest-value queries, so it never pays Proposition 14(e) and loses the history instead.

Proposition 11(d) says a hard machine spends none of the router's margin on order; Proposition 14(e) says what it spends instead.

Remark 17 (Exchange rates, measured). The proposition fixes the price; three registered tests measure how efficiently machines pay it (Section 3); a later amendment to the memory test only tightened its assessment.

- *Memory.* An outer-product (linear-attention) state of d^2 numbers, written with random unit keys in $\mathbb{R}^d$ and independent values uniform on $\{-1, +1\}^d$, and read by signs, recalls 0.315–0.328 nat per number, close to $1/\pi$, once it holds $16d$ items or more. A least-squares state, the delta rule's fixed point, recalls every stored bit exactly up to $N = d$, which is d^2 bits at $N = d$, and then falls to the same rate. The online delta rule with a unit update rate peaks at 0.272–0.275 nat per number at $N = d$ to $2d$, settles at 0.264–0.267 by $N = 64d$ and recalls what is recent.
- *Fuzzy addressing.* With noisy queries, identifying stored B-bit codes follows the exact binary-code law. With each query bit flipped independently with a fixed probability $0 \leq p < 1/2$, its cost per stored key tends to $\ln N/(\ln 2 - h(p))$ bits, set by the capacity $\ln 2 - h(p)$ of the binary symmetric channel. At the code lengths tested the law's slope runs up to about 6% above capacity, and the registered 10% tolerance on the measured slope failed at two of four flip rates.

- *Addressing positions.* For a unary feature (two dimensions), the largest logit range needed to put a majority on a target, among the positions checked, is about $0.78(n-1)^2$. A rotary code needs 5.7–25.6 nats up to $n = 4{,}096$ with $2\log_2 n$ dimensions, within 3.1 times the floor $\ln(n-1)$. Recency carried in the weights is unary whatever the code: an optimized rotary term never exceeds the bound of Proposition 11.
- *Ledger-specific failure.* On one recall task each machine fails at its own wall:
  - softmax at a fixed scale at its decoder limit, which moves with τ;
  - the argmax at the code limit;
  - an outer-product state with the question after at a memory wall that grows about linearly in d, 34–713 times earlier; a wider value code moves that wall and no other machine's.

  A one-slot machine reading the question first decides exactly as the argmax over the whole cache, by construction, with a state of one slot.

A.11 Anchored retrieval against fixed patterns

The gain-control certificate of Section 6 has a sharp form when the routing reads fixed stored patterns instead of the machine's own states, as modern Hopfield memory does (Appendix D). Take stored patterns $x_1, \dots, x_N \in \mathbb{R}^d$ of diameter $D = \max_{i,j}\|x_i - x_j\|_2$, biases b_i, a gain $\tau \geq 0$ (β in Ramsauer et al. (2021)), $0 \leq \alpha \leq 1$ and an anchor $a \in \mathbb{R}^d$, and let

$$T(q) = \sum_i p_i(q)\, x_i, \qquad p(q) = \operatorname{softmax}\big((\tau x_i^\top q + b_i)_i\big), \qquad F_a(q) = (1-\alpha)a + \alpha T(q),$$

so that the anchor is re-injected at every step and only the query moves. Write $\kappa = \alpha\tau D^2/4$.

Proposition 18 (Anchored retrieval)**.**

(a) The certificate. *F_a is κ-Lipschitz on $\mathbb{R}^d$. If $\|x_i - c\|_2 \leq R$ for every i and some point c, then $\alpha\tau R^2$ is also a Lipschitz constant.*

(b) Below the threshold. *If $\kappa < 1$, F_a has exactly one fixed point q_a, and every start q obeys $\|F_a^n(q) - q_a\|_2 \leq \kappa^n\|q - q_a\|_2$. Moreover $\|q - q_a\|_2 \leq \|q - F_a(q)\|_2/(1-\kappa)$ (so $\|q_a - a\|_2 \leq \alpha \max_i\|x_i - a\|_2/(1-\kappa)$), and two anchors give $\|q_a - q_{a'}\|_2 \leq (1-\alpha)\|a - a'\|_2/(1-\kappa)$.*

(c) At the threshold. *If $\kappa \leq 1$, F_a still has exactly one fixed point, and every orbit converges to it.*

(d) Sharpness. *For the two patterns $\pm M$ in $\mathbb{R}$ with $M > 0$, zero biases and $a = 0$, $F_a(q) = \alpha M \tanh(\tau M q)$ has Lipschitz constant exactly $\kappa = \alpha\tau M^2$. For $\kappa > 1$ it has three fixed points, and for $\kappa = 1$ no constant below one bounds it.*

With $\alpha = 1$ and zero biases there is no anchor, and the radius form with c the patterns' mean recovers the certificate of Ramsauer et al. (2021): $\tau m^2 < 1$, with m the largest distance of a pattern from that mean. Its unique fixed point is then the same for every query, so iteration forgets the query. The anchor scales the gain by α and keeps the query's identity: for $\alpha < 1$, two distinct anchors never share a fixed point, since subtracting their fixed-point equations leaves $(1-\alpha)(a - a') = 0$. The condition is sufficient, not necessary. A strong anchor can saturate the probabilities on the set that the iteration enters after one step, and keep the fixed point unique beyond $\kappa = 1$ (an example with $\kappa = 2$ follows the proof).

Proof of Proposition 18. Part (a). Softmax is unchanged by a common shift of its logits and is 1-Lipschitz from ℓ_∞ to ℓ_1 (Appendix A.5). Shifting one logit vector by the midpoint of the range of their difference gives

$$\|p(q) - p(r)\|_1 \leq \tfrac{1}{2}\Big(\max_i \tau x_i^\top (q-r) - \min_i \tau x_i^\top (q-r)\Big) \leq \tfrac{1}{2}\tau D\|q - r\|_2.$$

A zero-sum vector w has $\|\sum_i w_i x_i\|_2 \leq \frac{D}{2}\sum_i |w_i|$, as in the proof of Proposition 5(c). With $w = p(q) - p(r)$ this gives $\|T(q) - T(r)\|_2 \leq \frac{1}{4}\tau D^2\|q - r\|_2$, and the factor α gives κ. If every $\|x_i - c\|_2 \leq R$, then $D \leq 2R$.

Part (b). F_a is a κ-contraction of the complete space $\mathbb{R}^d$, so Banach's theorem gives the unique fixed point and the geometric bound. The triangle inequality and $F_a(q_a) = q_a$ give $\|q - q_a\| \le \|q - F_a(q)\| + \kappa\|q - q_a\|$. Subtracting the fixed-point equations of two anchors gives $\|q_a - q_{a'}\| \le (1-\alpha)\|a - a'\| + \kappa\|q_a - q_{a'}\|$.

Part (c). Every output of F_a lies in $K_a = (1-\alpha)a + \alpha\,\mathrm{conv}\{x_i\}$, a compact convex set that F_a maps into itself and that holds every fixed point. Since $\kappa \le 1$, part (a) shows that F_a does not increase distances. It decreases every distance between distinct points of K_a strictly. Suppose instead that $\|F_a(r) - F_a(q)\| = \|r - q\| > 0$ for some $q, r \in K_a$, and put $v = r - q$, $u = F_a(r) - F_a(q)$ and $\phi(t) = u^\top F_a(q + tv)$. For $0 \le t \le 1$, non-expansiveness and Cauchy–Schwarz give $\phi(t) - \phi(0) \le t\|v\|^2$ and $\phi(1) - \phi(t) \le (1-t)\|v\|^2$. Their sum is the equality $\phi(1) - \phi(0) = \|u\|^2 = \|v\|^2$, so $\phi(t) = \phi(0) + t\|v\|^2$ on $[0, 1]$. But ϕ is a constant plus a ratio of finite sums of exponentials of affine functions of t, with a positive denominator, so it is real analytic on the whole line and the identity holds for every real t. Its right side is unbounded, while ϕ is bounded because F_a takes values in the bounded set K_a. A continuous map of a nonempty compact set into itself that strictly decreases every distance has exactly one fixed point, which attracts every orbit: $\|q - F_a(q)\|$ attains its minimum on the set, and strict decrease makes that minimum zero; two fixed points would contradict strict decrease; and along an orbit the distance to the fixed point decreases to a limit that a convergent subsequence and strict decrease force to be zero. Every orbit enters K_a after one step, so the conclusion holds from every start.

Part (d). With patterns $\pm M$ and zero biases, $T(q) = M\tanh(\tau M q)$, so $F_a'(q) = \kappa\,\mathrm{sech}^2(\tau M q)$, whose maximum κ is attained at zero, and $D = 2M$. Writing $q = \alpha M y$ turns the fixed-point equation into $y = \tanh(\kappa y)$. For $\kappa > 1$, the function $\tanh(\kappa y) - y$ is positive just above zero, negative at $y = 1$ and strictly concave for $y > 0$, so it has exactly one positive root; with zero and the root's mirror image, there are three fixed points. At $\kappa = 1$ the derivative at zero is one, so no Lipschitz constant below one holds. □

Part (c) needs only that F_a does not increase distances on K_a, and that holds whenever $\alpha\tau\|C(q)\|_{\mathrm{op}} \le 1$ for every $q \in K_a$, where $C(q)$ is the covariance of the patterns under $p(q)$: the derivative of T is $\tau C(q)$. For the patterns ± 1, $\alpha = \frac{1}{2}$, $\tau = 4$ and $a = 4$, the map is $F_a(q) = 2 + \frac{1}{2}\tanh(4q)$ and $\kappa = 2$, but on $K_a = [1.5, 2.5]$ its derivative is at most $2\,\mathrm{sech}^2(6) < 1$, so the fixed point is unique.

B Other substrates: physical, chemical and biological machines

These statements read two conditions off physical substrates. They are statements about ideal devices, not measurements: convergence rates need specified capacitances and dynamics, and no device was simulated. Read onto a real physical, chemical or biological machine, each is a hypothesis, as are the catalog's entries for such machines (Appendix D).

Proposition 19 (A dynamic-range wall for one competitive stage)**.** *Let a stage normalize N positive conductances whose ratio of largest to smallest is at most R. Its target's weight is at most $R/(R + N - 1)$, so a target weight of at least $w \in (0, 1)$ needs $N \le 1 + R(1-w)/w$. The bound is sharp, so at fixed target weight and large R a stage buys about $\ln R$ nats of $\ln N$.*

Proof. Let $g_{\min}$ be the smallest of the conductances. The target's conductance is at most $Rg_{\min}$ and the other $N - 1$ are at least $g_{\min}$, and the target's share is largest at these extremes. Solving the resulting weight inequality gives the count. □

A shunt, the circuit's null, adds competing conductance and cannot raise the target's weight. Hard zeros, pruning and independent stages change the class: the bound applies to one stage, not to a whole cascaded device. It is the bound of Proposition 14(c), with the conductance ratio in the place of the exponentiated logit window, which is $e^{2\tau}$ for cosine scores at logit scale τ. So the on/off ratio of an ideal device caps the gain a single soft stage can spend.

Proposition 20 (A passive reciprocal circuit is a tethered router)**.** *Let $W = W^\top \ge 0$ hold reciprocal edge conductances, with zero diagonal and positive degrees $d_i = \sum_j W_{ij}$, and let $D = \mathrm{diag}(d_i)$. Connect node i to a clamped source at potential x_{0i} through a conductance $g_i = \beta d_i/\alpha$, with $0 < \alpha < 1$ and $\beta = 1 - \alpha$. Kirchhoff's steady-state equations have the unique solution $x = \beta(I - \alpha P)^{-1}x_0$ with $P = D^{-1}W$, which is the tethered*

fixed point of Proposition 4. For arbitrary positive shunts g_i, the solution is $[I - \mathrm{diag}(\alpha_i)P]^{-1}\,\mathrm{diag}(1-\alpha_i)x_0$ with $\alpha_i = d_i/(d_i + g_i)$.

Proof. Current balance at node i reads $\sum_j W_{ij}(x_i - x_j) + g_i(x_i - x_{0i}) = 0$. With proportional shunts this is $[(D - W) + (\beta/\alpha)D]x = (\beta/\alpha)Dx_0$, and multiplying by αD^{-1} gives the stated resolvent, whose inverse exists by the convergent Neumann series of αP, whose rows sum to $\alpha < 1$. For general shunts, dividing row i by $d_i + g_i$ gives the second form; its substochastic matrix has largest row sum $\max_i \alpha_i < 1$, which gives uniqueness in the same way. □

Reciprocal passive conductances make P reversible, with stationary weights proportional to the degrees d_i. A directed, nonreversible routing kernel needs further circuit elements. So, with proportional shunts, an ideal relaxation network computes the tethered router's fixed point exactly, and Proposition 4 applies to it unchanged; an arrow, if it needs one, must come from elements that break reciprocity. Analog crossbars already solve such linear systems, PageRank among them, in one step per right-hand side, though with operational amplifiers in feedback rather than a passive network (Sun et al., 2019), and nonlinear resistive networks have been trained end to end with equilibrium propagation in circuit simulation (Kendall et al., 2020).

C Experimental details

This appendix gives the protocols behind the measurements, test by test, and ends with a table that maps each test to its vendored records (Appendix C.8).

C.1 The constructed tests

The constructed tests use two substrates with $D = 128$. On the product substrate a memory of $N = B^2$ keys is formed from two codebooks of B unit vectors in $\mathbb{R}^{64}$ as $k_{ij} = (a_i, b_j)/\sqrt{2} \in \mathbb{R}^{128}$, and a key's score is its cosine with the query (the mean of its two half-cosines when the query halves have equal norm, as at $c = 1$); a present query has cosine exactly c with its target, with $c \in \{1, 0.5, 0.35\}$. On the isotropic substrate the keys are uniform random unit vectors in $\mathbb{R}^{128}$. Every test uses three seeds (five in the recovery's $N = 128$, $d = 32$ gain-control run), and each query-based point pools 2,000 queries per seed (1,000 on the isotropic substrate in the budget test, 500 in the separation test).

The machines are those of Sections 3 and 4. Softmax succeeds when the target holds at least half its mass. A select walker draws M samples and keeps the best one, and an averaging walker needs a strict majority of its samples; both success probabilities are evaluated exactly, conditional on each query. A hard tree takes the global argmax, and a soft tree reads a softmax over a $k \times k$ shortlist of half-argmaxes and needs half its mass on the target. Each null is calibrated at $N = 64$ on ten memories with 400 present and 400 absent queries: a fixed sink, the sink plus $\ln N - \ln 64$, a tempered sink that scales the content and sink logits by $\ln N/\ln 64$, 64 competitive walkers sharing the fixed or scheduled sink, 64 or N verifying walkers with a threshold, a sink inside the tree's 4×4 shortlist, and the tree's key check. The tether test routes the 256 keys of a 16×16 product memory as tokens carrying random unit cargo in $\mathbb{R}^{64}$, at $\tau = 10$ with self excluded. The gain-control test iterates the tethered softmax, 32-walker and hard-argmax maps from two random unit starts on $N = 64$ tokens in $\mathbb{R}^{16}$ ($N = 128$ in $\mathbb{R}^{32}$ in the recovery). The separation test (Section 7) scores tied keys as $\frac{1}{2}(z_j + g_{y_j})$, the mean of the address cosine z_j and the query's cosine g_{y_j} with one of 16 random unit cargo-class vectors, and the arrow test (Section 8) writes two identical copies of an item at random positions among $N = 1{,}024$ keys.

An onset is the first downward crossing of one half, interpolated linearly in $\ln N$. A curve that starts below one half has onset $-\infty$, and one that never crosses within its grid is right-censored, which is not an infinite capacity. The historical grid steps $\ln N$ by 1.39 nats; the recovery's fine grid steps it by 0.17 nats and contains the coarse points.

C.2 Recovery of the constructed tests

The constructed tests were recomputed from their sources: every recomputed historical field agrees with the historical package to within 5.3×10^{-15}, and the package is labelled not reproduced only because 17 effective ranks of the gain-control test differ by up to that amount under the registration's zero tolerance. The five constructed-test figures (Figures 2 and 5 to 8) and Table 6 use the accepted first recovery of the constructed tests: 14 cells that recompute the historical package and 36 new cells, under a frozen registration with its sources bound to it. Its saved assessment is a failure, because its execution check did not pass, and completeness does not change that. The registration has a branch for this case: when the port passes its own checks but differs from the historical records, the new records may be published as a rebaseline and are never called reproductions. That branch applies here, so the package's original values are historical records, not reproduced. Of the registered checks, the primary one, that every historical verdict keeps its status on the fine grid, fails: seven historical statuses change, all in the registered comparison of averaging walkers with softmax. The primary checks on the gain-control band, on the scheduled walkers' abstention and retrieval, and on the tether criterion's decode pass. A secondary check fails: the hard router was predicted to settle into a two-step oscillation at $\alpha = 0.7$, 0.9 and 0.95, and in 5 of 13 scored seed-α cases it settled into a fixed point instead. All plotted measurements and saved summaries are kept without reassessment, refitting or new uncertainty estimates.

The other 13 cells that recompute the historical package pass their checks: the 288 refinement fields and all 171 latest-write fields are identical, the 2,166 tether fields agree within 2.6×10^{-15}, and the defined budget, abstention and separation fields agree within 3.6×10^{-15}. The reproduction failure is in the recomputed gain-control test, at the hard router's effective rank. Of 3,240 compared fields, 2,335 are identical and 905 differ; 17 differences violate the registered zero tolerance. The rerun of that test at its recorded size is therefore compared with the run's own frozen recomputation, whose fields it equals exactly, and its disagreement with the historical record remains a diagnostic and is not concealed. A rounding defect in the reference walker code, a probability that exceeds one by rounding, left 18 fixture fields and 126 historical product-key endpoint fields undefined. Only these enumerated fields, each traced to the defect by a replay of the reference, are admitted with computed values, and they are not called reproduced. The original registration, gate closure, per-field comparisons and endpoint events remain in the vendored records.

Every failed field of the recomputed gain-control test is the effective rank of the hard router's state matrix, in one of the eight record rows of its sweep over $\alpha \in \{0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 0.99, 1\}$ (rows 0–7); their historical and recovered values are:

- seed0, row 0: $12.464954594672562 \to 12.464954594672566$, $|\Delta| = 3.55271 \times 10^{-15}$.
- seed0, row 1: $11.494796697675987 \to 11.494796697675989$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed0, row 5: $7.568130666698354 \to 7.5681306666983525$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed0, row 7: $8.59926233401497 \to 8.599262334014973$, $|\Delta| = 3.55271 \times 10^{-15}$.
- seed1, row 0: $12.36006662996179 \to 12.360066629961791$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed1, row 1: $11.414792011019198 \to 11.414792011019196$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed1, row 3: $9.539665495353841 \to 9.53966549535384$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed1, row 4: $9.227025588889033 \to 9.22702558888903$, $|\Delta| = 3.55271 \times 10^{-15}$.
- seed1, row 5: $9.116152294302989 \to 9.116152294302985$, $|\Delta| = 3.55271 \times 10^{-15}$.
- seed1, row 6: $9.345382447765608 \to 9.345382447765614$, $|\Delta| = 5.32907 \times 10^{-15}$.
- seed2, row 1: $11.691618697146337 \to 11.691618697146335$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed2, row 2: $10.719989999106332 \to 10.719989999106335$, $|\Delta| = 3.55271 \times 10^{-15}$.
- seed2, row 3: $10.479253538968194 \to 10.479253538968196$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed2, row 4: $10.053224982700028 \to 10.05322498270003$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed2, row 5: $9.99077570694618 \to 9.990775706946184$, $|\Delta| = 5.32907 \times 10^{-15}$.
- seed2, row 6: $9.94592152059316 \to 9.945921520593158$, $|\Delta| = 1.77636 \times 10^{-15}$.
- seed2, row 7: $9.874915320823858 \to 9.874915320823852$, $|\Delta| = 5.32907 \times 10^{-15}$.

Both registered tolerances are zero; the magnitude of a difference does not waive this gate.

Figure protocols. The product-key retrieval and abstention curves pool three seeds with 2,000 queries per seed (per query type for abstention). The fine grid extends to $\ln N = 19.408$, with the retained coarse endpoint at $N = 2^{30}$. No seed intervals survive pooling. Displayed onset summaries are the saved first downward 50% crossings, interpolated in $\ln N$; arrows retain censoring rather than manufacture finite onsets. The token-class figure (Figure 7) pools 544 cycle, 178 depth-one and 46 deeper tokens across three seeds at $N = 256$. The removal figure's identity paths retain all three seeds, $N = 256$, 64-dimensional random cargo, $\tau = 10$ and 64 walkers per token. Its gain panel retains each softmax start-pair distance at all four α values: $N = 64, d = 16$ has three seeds and a 1,500-update cap; $N = 128, d = 32$ has five seeds and a 5,000-update cap, with 1,500-step states saved. Either start hitting its cap is marked. Endpoint separation alone is not a certified comparison of fixed points. The separation panel uses $D = 128$ and three seeds with 500 queries each; softmax has $c = 1, \tau = 10, 20$, and the $M = 256, \tau = 10$ walker and hard argmax have $c = 0.35$ (historically registered at $c = 0.5$). The latest-write panel uses $N = 1024, D = 128$, three seeds with 2,000 queries each, and all 12 saved linear recency biases. Softmax success requires half the target mass and a 9:1 ratio over the older copy. The $M = 256$ walker has no recency in its sampling distribution; timestamp selection changes its read rule. Overwrite success is by construction. F^* and the best linear value are saved mean-field references conditional on retaining every position, not measured maxima of the plotted average-accuracy curve.

Certificate units and saved summaries. Table 6 is one coupled table in three continued blocks. It retains all 32 per-seed rows and the saved median, minimum and maximum for each of the eight size–α configurations in each block; these summaries are not recomputed. Entries are displayed to six significant figures; full values are in the source and render audit. τ_{loss} is the first grid point whose saved label is not unique, and τ_{last} is the preceding unique point. R_*, R_{traj} and $\widehat{\rho}$ are its realized fixed-point radius, maximum trajectory radius and estimated local gain. A capped endpoint is not relabelled as a fixed point. The tight threshold is $\tau_{\text{cert}}(R) = (1-\alpha)/(2\alpha R^2)$; the registered threshold has denominator $4\alpha R^2$. The dimensionless gain ratios are $q_{\text{reg}} = \tau_{\text{last}}/\tau_{\text{cert,reg}}$ and $q_{\text{tight}} = \tau_{\text{last}}/\tau_{\text{cert}}$; the separately named Lipschitz value is $\mathcal{L} = \alpha(1 + 2\tau_{\text{last}}R^2)$. Each is reported at $R = 1$ and $R = R_*$. The domain radius can define an invariant set; a measured fixed-point radius alone cannot. All quantities here are normalized and dimensionless. Minima/maxima are seed ranges, not confidence intervals.

Table 6: Gain-control transitions, radii and certificate diagnostics. Transitions and radii.

N	α	Seed / summary	τ_{loss}	τ_{last}	R_*	R_{traj}	$\widehat{\rho}$
64	0.3	0	16	11.3137	0.933443	1	0.825334
64	0.3	1	16	11.3137	0.933065	1	0.964413
64	0.3	2	22.6274	16	0.949193	1	0.97513
64	0.3	median	16	11.3137	0.933443	1	0.964413
64	0.3	min	16	11.3137	0.933065	1	0.825334
64	0.3	max	22.6274	16	0.949193	1	0.97513
64	0.5	0	11.3137	8	0.594852	1	0.654382
64	0.5	1	11.3137	8	0.597155	1	0.657912
64	0.5	2	11.3137	8	0.591447	1	0.646754
64	0.5	median	11.3137	8	0.594852	1	0.654382
64	0.5	min	11.3137	8	0.591447	1	0.646754
64	0.5	max	11.3137	8	0.597155	1	0.657912
64	0.7	0	22.6274	16	0.422526	1	0.854524
64	0.7	1	22.6274	16	0.424131	1	0.862201
64	0.7	2	22.6274	16	0.415089	1	0.8426
64	0.7	median	22.6274	16	0.422526	1	0.854524
64	0.7	min	22.6274	16	0.415089	1	0.8426
64	0.7	max	22.6274	16	0.424131	1	0.862201
64	0.9	0	32	22.6274	0.21644	1	0.916944
64	0.9	1	32	22.6274	0.211411	1	0.915846
64	0.9	2	32	22.6274	0.202998	1	0.915422
64	0.9	median	32	22.6274	0.211411	1	0.915846
64	0.9	min	32	22.6274	0.202998	1	0.915422
64	0.9	max	32	22.6274	0.21644	1	0.916944
128	0.3	0	16	11.3137	0.838145	1	0.886633
128	0.3	1	16	11.3137	0.863928	1	0.894102

Continued on next page

Table 6 (continued): Transitions and radii

N	α	Seed / summary	τ_{loss}	τ_{last}	R_*	R_{traj}	$\widehat{\rho}$
128	0.3	2	11.3137	8	0.75004	1	0.388696
128	0.3	3	16	11.3137	0.872525	1	0.829281
128	0.3	4	16	11.3137	0.877498	1	0.788384
128	0.3	median	16	11.3137	0.863928	1	0.829281
128	0.3	min	11.3137	8	0.75004	1	0.388696
128	0.3	max	16	11.3137	0.877498	1	0.894102
128	0.5	0	16	11.3137	0.557535	1	0.597363
128	0.5	1	16	11.3137	0.557755	1	0.594749
128	0.5	2	16	11.3137	0.556462	1	0.614619
128	0.5	3	16	11.3137	0.563097	1	0.607944
128	0.5	4	16	11.3137	0.555648	1	0.613803
128	0.5	median	16	11.3137	0.557535	1	0.607944
128	0.5	min	16	11.3137	0.555648	1	0.594749
128	0.5	max	16	11.3137	0.563097	1	0.614619
128	0.7	0	22.6274	16	0.348759	1	0.758803
128	0.7	1	22.6274	16	0.354194	1	0.761097
128	0.7	2	22.6274	16	0.352089	1	0.777326
128	0.7	3	22.6274	16	0.363485	1	0.763126
128	0.7	4	22.6274	16	0.342498	1	0.769982
128	0.7	median	22.6274	16	0.352089	1	0.763126
128	0.7	min	22.6274	16	0.342498	1	0.758803
128	0.7	max	22.6274	16	0.363485	1	0.777326
128	0.9	0	45.2548	32	0.154248	1	0.911809
128	0.9	1	45.2548	32	0.172521	1	0.910834
128	0.9	2	45.2548	32	0.166274	1	0.912665
128	0.9	3	45.2548	32	0.183164	1	0.911292
128	0.9	4	45.2548	32	0.153793	1	0.911275
128	0.9	median	45.2548	32	0.166274	1	0.911292
128	0.9	min	45.2548	32	0.153793	1	0.910834
128	0.9	max	45.2548	32	0.183164	1	0.912665

Table 6: Gain-control diagnostics (continued). Domain radius $R = 1$.

N	α	Seed / summary	τ_{cert} (tight)	q_{reg}	q_{tight}	$\mathcal{L}$
64	0.3	0	1.16667	19.3949	9.69746	7.08823
64	0.3	1	1.16667	19.3949	9.69746	7.08823
64	0.3	2	1.16667	27.4286	13.7143	9.9
64	0.3	median	1.16667	19.3949	9.69746	7.08823
64	0.3	min	1.16667	19.3949	9.69746	7.08823
64	0.3	max	1.16667	27.4286	13.7143	9.9
64	0.5	0	0.5	32	16	8.5
64	0.5	1	0.5	32	16	8.5
64	0.5	2	0.5	32	16	8.5
64	0.5	median	0.5	32	16	8.5
64	0.5	min	0.5	32	16	8.5
64	0.5	max	0.5	32	16	8.5
64	0.7	0	0.214286	149.333	74.6667	23.1
64	0.7	1	0.214286	149.333	74.6667	23.1
64	0.7	2	0.214286	149.333	74.6667	23.1
64	0.7	median	0.214286	149.333	74.6667	23.1
64	0.7	min	0.214286	149.333	74.6667	23.1
64	0.7	max	0.214286	149.333	74.6667	23.1
64	0.9	0	0.0555556	814.587	407.294	41.6294
64	0.9	1	0.0555556	814.587	407.294	41.6294
64	0.9	2	0.0555556	814.587	407.294	41.6294
64	0.9	median	0.0555556	814.587	407.294	41.6294

Continued on next page

Table 6 (continued): Domain radius $R = 1$

N	α	Seed / summary	τ_{cert} (tight)	q_{reg}	q_{tight}	$\mathcal{L}$
64	0.9	min	0.0555556	814.587	407.294	41.6294
64	0.9	max	0.0555556	814.587	407.294	41.6294
128	0.3	0	1.16667	19.3949	9.69746	7.08823
128	0.3	1	1.16667	19.3949	9.69746	7.08823
128	0.3	2	1.16667	13.7143	6.85714	5.1
128	0.3	3	1.16667	19.3949	9.69746	7.08823
128	0.3	4	1.16667	19.3949	9.69746	7.08823
128	0.3	median	1.16667	19.3949	9.69746	7.08823
128	0.3	min	1.16667	13.7143	6.85714	5.1
128	0.3	max	1.16667	19.3949	9.69746	7.08823
128	0.5	0	0.5	45.2548	22.6274	11.8137
128	0.5	1	0.5	45.2548	22.6274	11.8137
128	0.5	2	0.5	45.2548	22.6274	11.8137
128	0.5	3	0.5	45.2548	22.6274	11.8137
128	0.5	4	0.5	45.2548	22.6274	11.8137
128	0.5	median	0.5	45.2548	22.6274	11.8137
128	0.5	min	0.5	45.2548	22.6274	11.8137
128	0.5	max	0.5	45.2548	22.6274	11.8137
128	0.7	0	0.214286	149.333	74.6667	23.1
128	0.7	1	0.214286	149.333	74.6667	23.1
128	0.7	2	0.214286	149.333	74.6667	23.1
128	0.7	3	0.214286	149.333	74.6667	23.1
128	0.7	4	0.214286	149.333	74.6667	23.1
128	0.7	median	0.214286	149.333	74.6667	23.1
128	0.7	min	0.214286	149.333	74.6667	23.1
128	0.7	max	0.214286	149.333	74.6667	23.1
128	0.9	0	0.0555556	1152	576	58.5
128	0.9	1	0.0555556	1152	576	58.5
128	0.9	2	0.0555556	1152	576	58.5
128	0.9	3	0.0555556	1152	576	58.5
128	0.9	4	0.0555556	1152	576	58.5
128	0.9	median	0.0555556	1152	576	58.5
128	0.9	min	0.0555556	1152	576	58.5
128	0.9	max	0.0555556	1152	576	58.5

Table 6: Gain-control diagnostics (continued). Realized radius $R = R_*$.

N	α	Seed / summary	τ_{cert} (tight)	q_{reg}	q_{tight}	$\mathcal{L}$
64	0.3	0	1.33897	16.8991	8.44955	6.21468
64	0.3	1	1.34006	16.8854	8.44271	6.20989
64	0.3	2	1.2949	24.7122	12.3561	8.94928
64	0.3	median	1.33897	16.8991	8.44955	6.21468
64	0.3	min	1.2949	16.8854	8.44271	6.20989
64	0.3	max	1.34006	24.7122	12.3561	8.94928
64	0.5	0	1.41303	11.3232	5.66158	3.33079
64	0.5	1	1.40215	11.411	5.7055	3.35275
64	0.5	2	1.42935	11.1939	5.59696	3.29848
64	0.5	median	1.41303	11.3232	5.66158	3.33079
64	0.5	min	1.40215	11.1939	5.59696	3.29848
64	0.5	max	1.42935	11.411	5.7055	3.35275
64	0.7	0	1.20029	26.6602	13.3301	4.69904
64	0.7	1	1.19122	26.8631	13.4316	4.72947
64	0.7	2	1.24368	25.73	12.865	4.5595
64	0.7	median	1.20029	26.6602	13.3301	4.69904
64	0.7	min	1.19122	25.73	12.865	4.5595
64	0.7	max	1.24368	26.8631	13.4316	4.72947

Continued on next page

Table 6 (continued): Realized radius $R = R_*$

N	α	Seed / summary	τ_{cert} (tight)	q_{reg}	q_{tight}	$\mathcal{L}$
64	0.9	0	1.18591	38.1603	19.0802	2.80802
64	0.9	1	1.24301	36.4075	18.2037	2.72037
64	0.9	2	1.34817	33.5676	16.7838	2.57838
64	0.9	median	1.24301	36.4075	18.2037	2.72037
64	0.9	min	1.18591	33.5676	16.7838	2.57838
64	0.9	max	1.34817	38.1603	19.0802	2.80802
128	0.3	0	1.66077	13.6247	6.81234	5.06864
128	0.3	1	1.56312	14.4758	7.23792	5.36654
128	0.3	2	2.07386	7.7151	3.85755	3.00028
128	0.3	3	1.53247	14.7654	7.38269	5.46788
128	0.3	4	1.51515	14.9341	7.46707	5.52695
128	0.3	median	1.56312	14.4758	7.23792	5.36654
128	0.3	min	1.51515	7.7151	3.85755	3.00028
128	0.3	max	2.07386	14.9341	7.46707	5.52695
128	0.5	0	1.60852	14.0672	7.03362	4.01681
128	0.5	1	1.60725	14.0783	7.03917	4.01958
128	0.5	2	1.61473	14.0131	7.00657	4.00328
128	0.5	3	1.5769	14.3493	7.17465	4.08733
128	0.5	4	1.61946	13.9722	6.9861	3.99305
128	0.5	median	1.60852	14.0672	7.03362	4.01681
128	0.5	min	1.5769	13.9722	6.9861	3.99305
128	0.5	max	1.61946	14.3493	7.17465	4.08733
128	0.7	0	1.76174	18.1639	9.08193	3.42458
128	0.7	1	1.70809	18.7344	9.3672	3.51016
128	0.7	2	1.72857	18.5124	9.25619	3.47686
128	0.7	3	1.62189	19.7301	9.86506	3.65952
128	0.7	4	1.82674	17.5176	8.75879	3.32764
128	0.7	median	1.72857	18.5124	9.25619	3.47686
128	0.7	min	1.62189	17.5176	8.75879	3.32764
128	0.7	max	1.82674	19.7301	9.86506	3.65952
128	0.9	0	2.33501	27.4089	13.7045	2.27045
128	0.9	1	1.86656	34.2876	17.1438	2.61438
128	0.9	2	2.00947	31.8493	15.9246	2.49246
128	0.9	3	1.65595	38.6485	19.3242	2.83242
128	0.9	4	2.34884	27.2475	13.6238	2.26238
128	0.9	median	2.00947	31.8493	15.9246	2.49246
128	0.9	min	1.65595	27.2475	13.6238	2.26238
128	0.9	max	2.34884	38.6485	19.3242	2.83242

C.3 The evidence-capacity test and trained lookup heads

The evidence-capacity test draws explicit keys: Gaussian vectors in $\mathbb{R}^D$ whose cosine with a random unit query is the score, with the target at cosine c. It covers 20 settings, five dimensions at each of $c = 0.30$ ($D = 64$ to 192), 0.44 (32 to 80), 0.60 (16 to 40) and 0.80 (10 to 18), with 400 trials each in double precision. The argmax's onset is the logarithm of the median index of the first background key that outscores the target. Softmax's onsets are read at $\tau \in \{2.5, 5, 10, 20, 40, 80\}$, and the mean-field comparison is registered only inside the concentration regime. Uncertainties are bootstrap standard errors from 2,000 resamples. A separate cell draws a million keys at $D = 128$, over 100 trials, for the strongest background cosine.

The trained stage trains lookup heads on the noisy-match geometries of Bruce (2026c), with the match at cosine 0.6 or 0.8 and the key dimension varied, from 22 to 38 and from 11 to 17. Four rules, softmax, a sink, a log-N tempered sink and a key check, train at logit budgets $\tau_{\max} \in \{10, 20, 40\}$ with three seeds, 360 cells in all. Each head trains for 800 steps at 64 keys (batch 512, learning rate 0.05) and is evaluated at 16 to 65,536 keys, with accuracy and attention weight read separately.

C.4 The exchange-rate tests

The memory test writes N items into a state of $d \times d$ numbers, with $d \in \{16, 32, 64\}$ and N/d from $1/4$ to 64: an outer-product (linear-attention) state, a least-squares state and the online delta rule with a unit update rate. Keys are independent and uniform on the unit sphere, values are independent signs in each of d coordinates, and the read decides each coordinate by its sign. Each setting runs $\max(8, \lceil 2^{20}/(Nd) \rceil)$ trials. The information recalled per trial is a lower bound: the sum over age bins (of width $d/4$ up to age $8d$, then one bin) of the items in the bin times $d(\ln 2 - h(p_{\text{bin}}))$, with p_{bin} the bin's bit-error rate pooled over trials and coordinates. The rate divides it by d^2. An amendment capped a reported-only soft-read quantity after two exact least-squares cells overflowed; no scored quantity changed. The fuzzy-addressing part stores binary codes and queries them with each bit flipped independently with probability $p \in \{0, 0.05, 0.1, 0.2\}$, at five code lengths per p, with 400 trials per setting, trials censored at 50 times the law's onset for that setting, and 2,000 bootstrap resamples.

The positional-code test asks for a majority on any one of n positions, $n = 16$ to 4,096, with a one-hot code, a unary feature in two dimensions, and rotary codes with binary or geometric frequencies. The range is the spread of positional logits that serves every checked target: every target, except for the unary code above $n = 512$, where 262 targets are checked at each n (every $(n/256)$-th target, the ends and the midpoint). The rotary amplitudes are shared across targets and come from a convex program. Its recency part uses $n \in \{64, 256\}$, $K \in \{1, 2, 3, 4, 6\}$, $r = 9$ and range $K \ln 9$, and compares the staircase, linear biases, and a rotary term of six frequency pairs, the best of 32 optimizer restarts.

The ledger-failure test uses match cosine $c = 0.8$, 16 value classes, key dimensions $d \in \{12, 14, 16, 18\}$, value widths 32 and 128, and $\tau \in \{5, 10, 20\}$. Its machines are softmax over the cache at a fixed τ, the argmax over the cache, an outer-product state that reads the question last, and a one-slot maximum that reads it first, with 400 trials per setting (4,000 for the state).

C.5 Small trained machines

Every mixer sits in the same pre-normalized block template, $x \leftarrow x + \text{Mixer}(\text{LN}\, x)$ followed by $x \leftarrow x + \text{MLP}(\text{LN}\, x)$ with a GELU MLP of width $2d$, and every mixer has its own query, key, value and output maps. The vision transformer has width 64, four heads and four layers, and reads 4×4 patches of 32×32 CIFAR-10 images: 65 tokens with the class token, and 145 at 48 pixels, where the position table is resized bilinearly. It trains for 1,170 steps of 128 images with random horizontal flips. The language model has width 128, four heads, four layers and a byte vocabulary, with a causal depthwise convolution of width 4 at its input and no positional encoding. It trains for 1,500 steps of 32 windows of 128 bytes of WikiText-2. Both use AdamW (learning rate 2×10^{-3}, weight decay 0.05), a one-cycle schedule and gradient clipping at 1.0. The walker draws $M = 8$ samples on images and 16 on language; the tree selects 4 blocks of 2×2 patches (the class token is always visible) or 2 blocks of 16 bytes, beside the query's own block; the address book uses two rounds of eight buckets; the relaxation network sums nine terms of its resolvent, with a continuation learned per head; the market runs three Sinkhorn iterations; the binding rule on language is a sigmoid of the score plus a bias learned per head minus $\ln n_i$, with a norm on its output; and the bounded state uses positive features with a decay learned per head. The log-N scale learned in training multiplies each logit by $s \ln n_i$, with n_i the keys visible to query i and s learned per layer and head, initialized so that the factor is one at 128 keys.

The language models are evaluated on every non-overlapping 513-byte window of the validation split (2,234 windows) with the full context, in 128-byte chunks, and with a sliding window that always sees 97 to 128 bytes. Δ_L is the full-context loss minus the chunked loss over positions 128 to $L - 1$, in bits per byte. The unit of every statistic is the seed. Each estimate has a percentile-bootstrap interval (10,000 resamples) and a Student-t interval, contrasts are paired by seed, a positive effect is resolved when both 95% lower bounds exceed zero, and Δ_{512} is scored as non-positive when both 95% upper bounds are at most $\delta = 0.004$ bits per byte. The registered small-model prediction is refuted. The tables report the accepted first recovery run without recomputing its assessment, and the historical pilot is kept separate. In the metric names, `evallogn` denotes evaluation-only log-N scaling (multiplying logits by $\ln 145/\ln 65$ at 48 pixels and by $\max(1, \ln(i+1)/\ln 128)$ at language position i), `ssmax` denotes training-time log-N scaling, and `prop` denotes the walker's evaluation rule $M_i = \min(n_i, \max(M_0, \lceil M_0 n_i / L_0 \rceil))$ ($M = 18$ at 48 pixels and M_i rising from 17

to 64 over language positions 128 to 511). The primary transfer is full minus chunked loss; `delta_sw512` instead uses the sliding-window comparison. In Table 10, image parity contrasts are accuracy fractions (so 5 and 3 percentage points are 0.05 and 0.03), and `evallogn_gain` is $\mathrm{acc}_{48,\mathrm{evallogn}} - \mathrm{acc}_{48}$ on images and $\Delta_{512} - \Delta_{512,\mathrm{evallogn}}$ on language (negative when evaluation scaling increases the length penalty).

Table 7: Registered trained machines, parameter counts and scored seeds. The length classes were fixed before the runs.

Task / mixer / training	Family	Length class	Parameters	Seeds
image / linear / plain	bounded state (linear)	length rule	140954	5
image / lsh / plain	address book (LSH)	competition	140938	5
image / resolvent / plain	relaxation (resolvent)	length rule	140954	5
image / sinkhorn / plain	market (Sinkhorn)	none	140938	5
image / softmax / plain	softmax (the vertex)	competition	140938	5
image / tree / plain	tree (top-k blocks)	competition	140938	5
image / walker / plain	walker (M sampled)	competition	140938	5
lm / linear / plain	bounded state (linear)	length rule	594320	10
lm / lsh / plain	address book (LSH)	competition	594304	10
lm / resolvent / plain	relaxation (resolvent)	length rule	594320	10
lm / sigmoid / plain	binding (sigmoid -ln n)	length rule	594448	10
lm / softmax / plain	softmax (the vertex)	competition	594304	10
lm / softmax / ssmax	softmax (the vertex)	competition	594320	10
lm / tree / plain	tree (top-k blocks)	competition	594304	10
lm / tree / ssmax	tree (top-k blocks)	competition	594320	10
lm / walker / plain	walker (M sampled)	competition	594304	10
lm / walker / ssmax	walker (M sampled)	competition	594320	10

Table 8: Saved seed-level means and 95% intervals. Accuracy is a fraction; drops and gains are signed differences. Language-model quantities are bits per byte. Bootstrap: 10,000 seed resamples, seed 0. The t intervals are the saved Student-t guard intervals.

Metric	Seeds	Mean	Bootstrap 95% CI	Student-t 95% CI
image / linear / plain				
acc32	5	0.57714	[0.56694, 0.58858]	[0.56015, 0.59413]
acc48	5	0.52006	[0.50432, 0.53584]	[0.49480, 0.54532]
acc48_evallogn	5	0.52006	[0.50432, 0.53584]	[0.49480, 0.54532]
drop	5	-0.05708	[-0.06892, -0.04764]	[-0.07401, -0.04015]
evallogn_gain	5	0.00000	[0.00000, 0.00000]	[0.00000, 0.00000]
image / lsh / plain				
acc32	5	0.55844	[0.55610, 0.56106]	[0.55455, 0.56233]
acc48	5	0.52232	[0.51584, 0.53043]	[0.51083, 0.53381]
acc48_evallogn	5	0.52560	[0.51896, 0.53186]	[0.51566, 0.53554]
drop	5	-0.03612	[-0.04312, -0.02986]	[-0.04624, -0.02600]
evallogn_gain	5	0.00328	[0.00082, 0.00624]	[-0.00097, 0.00753]
image / resolvent / plain				
acc32	5	0.54900	[0.54482, 0.55318]	[0.54246, 0.55554]
acc48	5	0.51538	[0.50672, 0.52140]	[0.50322, 0.52754]
acc48_evallogn	5	0.51794	[0.50646, 0.52572]	[0.50198, 0.53390]
drop	5	-0.03362	[-0.04486, -0.02533]	[-0.04971, -0.01753]
evallogn_gain	5	0.00256	[-0.00048, 0.00562]	[-0.00261, 0.00773]
image / sinkhorn / plain				
acc32	5	0.60892	[0.60124, 0.61698]	[0.59621, 0.62163]
acc48	5	0.56470	[0.56168, 0.56744]	[0.56016, 0.56924]
acc48_evallogn	5	0.56734	[0.56314, 0.57208]	[0.56021, 0.57447]

Metric	Seeds	Mean	Bootstrap 95% CI	Student-t 95% CI
drop	5	-0.04422	[-0.05296, -0.03720]	[-0.05685, -0.03159]
evallogn_gain	5	0.00264	[0.00044, 0.00522]	[-0.00139, 0.00667]
image / softmax / plain				
acc32	5	0.60180	[0.59856, 0.60480]	[0.59673, 0.60687]
acc48	5	0.55770	[0.55260, 0.56304]	[0.54921, 0.56619]
acc48_evallogn	5	0.56158	[0.55470, 0.56846]	[0.55050, 0.57266]
drop	5	-0.04410	[-0.05080, -0.03740]	[-0.05498, -0.03322]
evallogn_gain	5	0.00388	[0.00026, 0.00856]	[-0.00279, 0.01055]
image / tree / plain				
acc32	5	0.58276	[0.58018, 0.58500]	[0.57890, 0.58662]
acc48	5	0.54166	[0.52944, 0.55262]	[0.52429, 0.55903]
acc48_evallogn	5	0.54140	[0.52750, 0.55388]	[0.52163, 0.56117]
drop	5	-0.04110	[-0.05084, -0.03236]	[-0.05477, -0.02743]
evallogn_gain	5	-0.00026	[-0.00188, 0.00142]	[-0.00279, 0.00227]
image / walker / plain				
acc32	5	0.56696	[0.56130, 0.57252]	[0.55798, 0.57594]
acc48	5	0.52394	[0.51602, 0.53186]	[0.51120, 0.53668]
acc48_evallogn	5	0.53052	[0.52248, 0.53750]	[0.51876, 0.54228]
acc48_prop	5	0.53218	[0.52186, 0.54164]	[0.51650, 0.54786]
drop	5	-0.04302	[-0.05280, -0.03316]	[-0.05895, -0.02709]
evallogn_gain	5	0.00658	[0.00452, 0.00770]	[0.00374, 0.00942]
lm / linear / plain				
bpb128	10	2.13992	[2.13786, 2.14203]	[2.13734, 2.14250]
delta256	10	-0.03223	[-0.03249, -0.03197]	[-0.03254, -0.03191]
delta512	10	-0.03266	[-0.03287, -0.03249]	[-0.03289, -0.03243]
delta_sw512	10	0.00003	[-0.00004, 0.00012]	[-0.00007, 0.00013]
lm / lsh / plain				
bpb128	10	2.25953	[2.25816, 2.26085]	[2.25791, 2.26115]
delta256	10	-0.00567	[-0.00599, -0.00540]	[-0.00602, -0.00531]
delta512	10	0.00529	[0.00481, 0.00579]	[0.00471, 0.00588]
delta_sw512	10	0.02249	[0.02193, 0.02306]	[0.02181, 0.02318]
delta512_evallogn	10	0.00464	[0.00394, 0.00531]	[0.00381, 0.00547]
lm / resolvent / plain				
bpb128	10	2.26329	[2.26124, 2.26544]	[2.26074, 2.26585]
delta256	10	-0.01220	[-0.01266, -0.01182]	[-0.01271, -0.01168]
delta512	10	-0.00703	[-0.00765, -0.00652]	[-0.00773, -0.00634]
delta_sw512	10	0.01133	[0.01090, 0.01172]	[0.01083, 0.01183]
delta512_evallogn	10	-0.00635	[-0.00701, -0.00582]	[-0.00709, -0.00562]
lm / sigmoid / plain				
bpb128	10	2.27957	[2.27774, 2.28185]	[2.27706, 2.28207]
delta256	10	-0.00993	[-0.01053, -0.00942]	[-0.01061, -0.00925]
delta512	10	-0.00252	[-0.00296, -0.00209]	[-0.00304, -0.00199]
delta_sw512	10	0.01592	[0.01556, 0.01630]	[0.01547, 0.01637]
delta512_evallogn	10	0.00039	[-0.00032, 0.00110]	[-0.00047, 0.00125]
lm / softmax / plain				
bpb128	10	2.24029	[2.23801, 2.24286]	[2.23732, 2.24326]
delta256	10	-0.00039	[-0.00100, 0.00028]	[-0.00116, 0.00039]
delta512	10	0.01023	[0.00916, 0.01114]	[0.00903, 0.01142]
delta_sw512	10	0.02376	[0.02270, 0.02461]	[0.02261, 0.02490]
delta512_evallogn	10	0.01245	[0.01104, 0.01368]	[0.01084, 0.01406]
lm / softmax / ssmax				
bpb128	10	2.23726	[2.23429, 2.24041]	[2.23352, 2.24100]
delta256	10	-0.00039	[-0.00113, 0.00026]	[-0.00122, 0.00044]
delta512	10	0.01313	[0.01175, 0.01430]	[0.01159, 0.01466]
delta_sw512	10	0.02871	[0.02739, 0.02986]	[0.02723, 0.03019]
lm / tree / plain				
bpb128	10	2.23479	[2.23233, 2.23778]	[2.23146, 2.23812]
delta256	10	0.00151	[0.00083, 0.00209]	[0.00075, 0.00228]

Metric	Seeds	Mean	Bootstrap 95% CI	Student-t 95% CI
delta512	10	0.01101	[0.01007, 0.01189]	[0.00991, 0.01210]
delta_sw512	10	0.02197	[0.02121, 0.02267]	[0.02109, 0.02285]
delta512_evallogn	10	0.01638	[0.01522, 0.01745]	[0.01504, 0.01772]
lm / tree / ssmax				
bpb128	10	2.23177	[2.22895, 2.23517]	[2.22801, 2.23552]
delta256	10	0.00533	[0.00452, 0.00606]	[0.00439, 0.00627]
delta512	10	0.02018	[0.01899, 0.02131]	[0.01878, 0.02159]
delta_sw512	10	0.03201	[0.03097, 0.03309]	[0.03074, 0.03328]
lm / walker / plain				
bpb128	10	2.25054	[2.24819, 2.25313]	[2.24752, 2.25356]
delta256	10	0.00507	[0.00387, 0.00607]	[0.00375, 0.00639]
delta512	10	0.01984	[0.01802, 0.02137]	[0.01779, 0.02189]
delta_sw512	10	0.03072	[0.02930, 0.03195]	[0.02910, 0.03233]
delta512_evallogn	10	0.02147	[0.01949, 0.02313]	[0.01929, 0.02366]
delta512_prop	10	0.00727	[0.00570, 0.00863]	[0.00550, 0.00905]
lm / walker / ssmax				
bpb128	10	2.24760	[2.24424, 2.25086]	[2.24360, 2.25159]
delta256	10	0.00204	[0.00085, 0.00318]	[0.00064, 0.00345]
delta512	10	0.01888	[0.01712, 0.02069]	[0.01672, 0.02104]
delta_sw512	10	0.03388	[0.03227, 0.03562]	[0.03185, 0.03591]
delta512_prop	10	0.00906	[0.00726, 0.01101]	[0.00679, 0.01133]

Table 9: The registered structural prediction is refuted. Each entry is the saved assessment outcome; tree readings are reported separately. The first row is a compound prediction: it is not met because softmax kept its own penalty under log-N training, while the tree kept its penalty.

Prediction	Saved registered outcome
Under log-N training the tree keeps its penalty while softmax loses its own	not met
No length penalty with a length rule: linear	holds
No length penalty with a length rule: resolvent	holds
No length penalty with a length rule: sigmoid	holds
A length penalty under competition: lsh	holds
A length penalty under competition: softmax	holds
A length penalty under competition: tree	holds
A length penalty under competition: walker	holds
Log-N training removes the penalty: softmax	not met
Log-N training removes the penalty: tree	not met
Log-N training removes the penalty: walker	not met
The fixed-sample walker degrades most	holds
Samples growing with context cut the walker's penalty	holds
Assessment	refuted

Table 10: Historical predictions rescored on the registered run, with the historical package verdict in each group heading. Means and both 95% intervals are copied from the saved assessment. The two parity predictions show machine-minus-softmax contrasts and their saved tolerance-band tests; the bounded-competition prediction shows the named loss contrasts; the resolution prediction shows resolution drops and evaluation-scaling gains (accuracy fractions). No new overall verdict is assigned to the historical predictions.

Contrast or metric	Mean	Bootstrap 95% CI	Student-t 95% CI	In band
Parity: every machine within 5 points or 0.10 bits per byte of softmax (historical package: partial)				
image / linear	-0.02466	[-0.03266, -0.01502]	[-0.03857, -0.01075]	yes
image / lsh	-0.04336	[-0.04842, -0.03716]	[-0.05222, -0.03450]	yes

Contrast or metric	Mean	Bootstrap 95% CI	Student-t 95% CI	In band
image / resolvent	-0.05280	[-0.05538, -0.04998]	[-0.05708, -0.04852]	no
image / sinkhorn	0.00712	[-0.00070, 0.01494]	[-0.00567, 0.01991]	yes
image / tree	-0.01904	[-0.02396, -0.01556]	[-0.02611, -0.01197]	yes
image / walker	-0.03484	[-0.03864, -0.03124]	[-0.04072, -0.02896]	yes
lm / linear	-0.10037	[-0.10202, -0.09861]	[-0.10243, -0.09831]	no
lm / lsh	0.01925	[0.01673, 0.02135]	[0.01643, 0.02206]	yes
lm / resolvent	0.02301	[0.02173, 0.02426]	[0.02148, 0.02454]	yes
lm / sigmoid	0.03928	[0.03684, 0.04159]	[0.03640, 0.04216]	yes
lm / tree	-0.00550	[-0.00674, -0.00438]	[-0.00694, -0.00406]	yes
lm / walker	0.01025	[0.00893, 0.01169]	[0.00858, 0.01192]	yes
Walker and tree within 3 points or 0.08 bits per byte of softmax (historical package: confirmed)				
image / tree	-0.01904	[-0.02396, -0.01556]	[-0.02611, -0.01197]	yes
image / walker	-0.03484	[-0.03864, -0.03124]	[-0.04072, -0.02896]	no
lm / tree	-0.00550	[-0.00674, -0.00438]	[-0.00694, -0.00406]	yes
lm / walker	0.01025	[0.00893, 0.01169]	[0.00858, 0.01192]	yes
Bounded competition degrades less with length (historical package: refuted)				
linear-resolvent	-0.02563	[-0.02619, -0.02496]	[-0.02638, -0.02487]	–
linear-softmax	-0.04289	[-0.04378, -0.04178]	[-0.04411, -0.04166]	–
lsh-resolvent	0.01233	[0.01179, 0.01294]	[0.01162, 0.01303]	–
lsh-softmax	-0.00493	[-0.00574, -0.00417]	[-0.00587, -0.00399]	–
resolvent / evallogn_gain	-0.00068	[-0.00078, -0.00058]	[-0.00081, -0.00056]	–
softmax / evallogn_gain	-0.00223	[-0.00272, -0.00170]	[-0.00285, -0.00160]	–
tree-resolvent	0.01804	[0.01706, 0.01908]	[0.01682, 0.01927]	–
tree-softmax	0.00078	[-0.00004, 0.00164]	[-0.00025, 0.00182]	–
walker-resolvent	0.02687	[0.02515, 0.02852]	[0.02482, 0.02893]	–
walker-softmax	0.00961	[0.00809, 0.01094]	[0.00787, 0.01135]	–
Resolution transfer and evaluation-time log-N scaling (historical package: partial)				
linear / drop	-0.05708	[-0.06892, -0.04764]	[-0.07401, -0.04015]	–
linear / evaluation gain	0.00000	[0.00000, 0.00000]	[0.00000, 0.00000]	–
lsh / drop	-0.03612	[-0.04312, -0.02986]	[-0.04624, -0.02600]	–
lsh / evaluation gain	0.00328	[0.00082, 0.00624]	[-0.00097, 0.00753]	–
Positive logit-mixer seed means under evaluation scaling: 5				
resolvent / drop	-0.03362	[-0.04486, -0.02533]	[-0.04971, -0.01753]	–
resolvent / evaluation gain	0.00256	[-0.00048, 0.00562]	[-0.00261, 0.00773]	–
sinkhorn / drop	-0.04422	[-0.05296, -0.03720]	[-0.05685, -0.03159]	–
sinkhorn / evaluation gain	0.00264	[0.00044, 0.00522]	[-0.00139, 0.00667]	–
softmax / drop	-0.04410	[-0.05080, -0.03740]	[-0.05498, -0.03322]	–
softmax / evaluation gain	0.00388	[0.00026, 0.00856]	[-0.00279, 0.01055]	–
tree / drop	-0.04110	[-0.05084, -0.03236]	[-0.05477, -0.02743]	–
tree / evaluation gain	-0.00026	[-0.00188, 0.00142]	[-0.00279, 0.00227]	–
tree minus softmax drop	0.00300	[-0.00552, 0.01152]	[-0.01048, 0.01648]	–
walker / drop	-0.04302	[-0.05280, -0.03316]	[-0.05895, -0.02709]	–
walker / evaluation gain	0.00658	[0.00452, 0.00770]	[0.00374, 0.00942]	–

C.6 The readout battery's first stage

The first stage reads all 135 checkpoints of Appendix C.5: ten seeds of each of the ten language-model groups (seven mixers, three of them also trained with the log-N scale) and five seeds of each of the seven image mixers. For the language models a repeated-segment probe places a 16-byte segment's eight-byte prefix at the end of filler text and asks for its ninth byte, with the full segment planted earlier or absent. A head qualifies as a retrieval head if its mean weight on the two target positions (the prefix end and the ninth byte) is at least 0.3 across 128 identification trials at the 128-byte training context, a threshold fixed before scoring, and a prediction with fewer than three seeds that contribute a decided head is inconclusive. Images have no recall probe. The tether readout iterates each frozen routing matrix on random unit cargo, without a tether, and compares identity retention with spread in the three hard-routed image mixers against softmax. The gain readout covers the four mixers whose single step is a softmax over scores (softmax, tree, address book

and walker, the walker under both sampling rules), in 5 image and 9 language-model groups. It evaluates the certificate's quantity $\kappa = \hat{\alpha}(1 + 2\tau R^2)$ for each head, with $\hat{\alpha}$ the layer's measured share of the residual update and τR^2 the norm of the head's query–key map times the squared radius of its LayerNorm rows, as if the head were iterated in LayerNorm coordinates with any hard-selected support held fixed. The certificate needs $\kappa < 1$; the prediction's band is 0.1.

C.7 The max-statistic head in 125M language models

The max-statistic head was trained in the harness of Bruce (2026c): 125M-parameter language models with 12 layers of width 768, 12 heads, rotary positions and a 1,024-token context, trained for 9,537 steps of 256 sequences (about 2.5 billion tokens), with three seeds per rule. The head routes each query by a softmax over its eight largest causal content logits (all of them before the eighth position), with ties going to the earlier position, and gates its output by $g_i = \sigma(a(\max_{j<i} s_{ij} - \theta))$, reading s_{00} at the first position, with a positive and θ learned per head, initialized at slope one and threshold zero, and exempt from weight decay. It has no sink, so its null weight is $A = 1 - g_i$, and no length schedule. Every rule is evaluated to 8 times the training length, with native positions and with YaRN (Peng et al., 2024), which scales the routing logits while the gate reads the scores before YaRN's temperature. No head reached the harness's registered threshold for a retrieval head (a mean weight of 0.3 on the matching fact over 64 identification trials at the training length), so the probes read 3 fallback heads per model, with 64 probe trials per point. The registered secondary prediction asks that the mean per-seed changes in the log-odds of the absent-query null weight A and the present-query match weight W stay above -0.5 from the training length to 8 times it. With YaRN it holds on seed means, and jointly in 2 of 3 seeds (Section 11); with native positions it fails, with A's log-odds falling by 2.98 on average and the match weight reaching zero at 8 times the training length in every seed.

C.8 Records

Table 11 names each test and the file or folder of the package's `data/` that holds its records. The package manifest lists every file with its SHA-256 checksum, and every folder except the four of historical records also carries its own. The drivers and registrations of the registered tests are in `experiment_code/`, under the same folder names, or with their records in `data/`; the 125M harness's driver is identified by its hash in `data/SX02/SOURCES.json` and released with Bruce (2026c).

Test or record	Where it is used	Records in `data/`
The illustrative constructions of Figure 1	Figure 1	`hero_construction.json`
The historical constructed-test package: registration and records (code in `experiment_code/`)	The historical paragraphs of Sections 3 to 8; Appendix C.2	`lattice_machines_package/`
Historical product-key retrieval records	Section 3; Figure 3b	`budget_exchange/`
Historical abstention records	Section 4	`abstention/`
Historical tether records	Section 5	`tether/`
Historical gain-control, separation and arrow records	Sections 6 to 8	`removal_tests/`
Numerical evaluation of the contamination law	Section 7	`separation_quadrature/`
The constructed-test recovery	Figures 2 and 5 to 8, Table 6, Sections 3 to 8, Appendix C.2	`SX26/`
The evidence-capacity test	Section 3; Figure 3a	`SX29/`
Trained lookup heads	Section 3	`SX29_T/`
The memory test	Section 3; Figure 4b	`SX57/`
The positional-code test	Section 3; Figure 4c,d	`SX58/`
The ledger-failure test	Section 3; Figure 4a	`SX59/`
The small trained tests	Section 10; Figure 9; Tables 7 to 10	`SX27/`
The readout battery's first stage	Section 11; Table 4	`SX28/`
The max-statistic head in 125M language models	Section 11; Figures 1 and 10	`SX02/`

Table 11: The package's records, by test. Each entry holds the vendored records behind the named results, all listed in the package manifest; registered tests also have their drivers and registrations, under `experiment_code/` or with their records.

D The catalog of lattice machines

This appendix lists the 43 families of Section 9 by group (Table 12). For each it gives how the family meets the conditions, with the works that establish each point; where its writer sits, for the entries that name one; what predicts its failure; and its label. A *proposition* entry lies inside the class that a proposition is stated for. A *reading* follows by the same computation as a proposition, or is an interpretation, and is untested; a proposition named outside its class is marked "reading". A *hypothesis* is a physical or biological system whose reading as a machine that routes by content is ours, and an *analogy* is a social one. "Ours" marks a prediction that the cited works do not make, and a star marks the next tests of Section 9.

Table 12: The catalog, family by family.

Family	How it pays, and its writer	Failure predicted by	Label
Soft routers that compete for a conserved mass			
Softmax attention, the reference	*Tether:* the residual stream. *Budget:* logit gain; scaling the logits with $\ln n$ helps it transfer across lengths (Chiang & Cholak, 2022; Nakanishi, 2025). *Null:* attention sinks, which compete with every key (Xiao et al., 2024); registers are framed as storage for global information, not as abstention (Darcet et al., 2024). Two recent designs let the null grow with length: a learned tempering of every score, with the null raised like the background's extreme value (Akbar, 2026), and a sink calibrated to the corpus size in training (Gollapudi et al., 2026). *Arrow:* asymmetric scores or a causal mask (Vaswani et al., 2017). *Writer:* the destination MLP, which acts as a key–value memory (Geva et al., 2021).	The gain wall (Proposition 2); a competitive null needs $\ln N$, and the two-sided law follows (Proposition 3(a)); a logit range of $\ln(N-1)$ (Proposition 14(c)); the evidence wall (Proposition 1); a soft arrow's price (Proposition 11).	Proposition
Sigmoid attention	Each key binds independently, in the form of Langmuir's isotherm (Langmuir, 1918), so nothing competes and the total weight grows like n unless a bias of $-\ln n$ is added (Ramapuram et al., 2025). The source presents that bias, and a hybrid normalization it needed only at $n = 4096$ and 7B parameters, as norm control for stable training, not as a retrieval budget. *Writer:* the destination MLP. Ours: with that bias, abstention is length-invariant, but one key's weight falls like $1/N$ unless match logits grow like $\ln N$; the source has no test of retrieval against length.	Outside the classes of Proposition 3 and Proposition 14(c): no normalizer. By the same computation the bias is a $\ln N$ null schedule: one key's weight $\sigma(\tau c - \ln N)$ needs $\tau c \gtrsim \ln N$ (reading). The evidence wall (Proposition 1).	Reading
Stick-breaking attention*	Keys are scanned from nearest to farthest, and each takes a fraction $\sigma(s_j)$ of the stick that remains (Tan et al., 2025). Weights sum to at most one, so the leftover is a built-in null (an explicit "attend to nothing"), and the scan's order is a built-in arrow; the model uses no position embeddings, and larger models learn a sink-like remainder embedding. *Budget:* with equal background logits the leftover is $(1-\sigma(s))^N$, which vanishes as N grows unless background logits carry about $-\ln N$; that bias is ours, and the source neither uses nor discusses one. *Writer:* the destination MLP.	Outside the classes of Proposition 3 and Proposition 11. The log of the leftover sums over all N keys, so the law of sums applies by the same computation (reading). Order is bought with match logit, not positional range: the nearer of two identical writes gets at least $1/(1-\sigma(s))$ times the farther's weight.	Reading

Family	How it pays, and its writer	Failure predicted by	Label
Sinkhorn attention*	Rows and columns are both normalized, so every key must receive mass (Sander et al., 2022; Cuturi, 2013), and abstention fails on both axes without dustbins. SuperGlue's dustbins are sized to the other side (Sarlin et al., 2020). Sinkformers normalize only in the encoder, because of the causal mask; indeed a lower-triangular doubly stochastic matrix is the identity, so exact two-sided causal normalization degenerates. *Writer:* the destination MLP. Ours: without a dustbin irrelevant keys still receive mass. A dustbin pays in capacity, not in logit. One of fixed capacity c holds at most c in total, so k absent queries keep on average at most c/k of their mass, whatever the scores. One sized to the other side holds an absent query's share as N grows: with all other scores equal and dustbin score α, an exact construction gives $\sigma(\alpha/2)$ at every N. A match's weight still falls like $1/N$ at a fixed margin.	Outside the class of Proposition 3(a), which is stated for one normalizer: the dustbin's budget is its capacity (reading); the gain wall (Proposition 2; reading).	Reading
Slot attention	The softmax runs over K slots for each input, so every input must go somewhere; there is no dedicated background slot, and the background spreads over all slots (Locatello et al., 2020). Inputs are then pooled by a weighted mean, a second normalization. A recurrent update refines each slot, and more test-time iterations (up to seven) helped, with no slot swapping reported. *Writer:* the recurrent update and a per-slot MLP. Ours: enough iterations at enough sharpness make slots swap.	Multistability of the kind Proposition 5 rules out (reading: the map is outside its class). A background slot would be a competitive null over K slots, so its own share of each input needs no length schedule. But each slot's weighted mean is a second normalization, over the N inputs: an object slot stays pure only if each background input's weight on it falls like $1/N$, which needs a logit gap that grows like $\ln N$ (reading).	Reading
Capsule routing by agreement	Routing coefficients are rebuilt from the output at every iteration (Sabour et al., 2017; Hinton et al., 2018). Agreement accumulates in the logits with no pull-back: the prior logits are only the starting point, not a tether, and a "none-of-the-above" category acts as a routing null. Two to three iterations work best and more are slightly worse, which Paik et al. (2019) trace to an uncontrolled feedback loop in which the logits grow without bound; lowering the routing gain helps. *Writer:* the capsule transforms, at the source of each part–whole pair. Ours: re-injecting the prior at every iteration would let more iterations help.	An effective gain that rises with every iteration, of the kind Proposition 5 bounds (reading: outside its class).	Reading
Stochastic and sampling machines			
Restart walker	Each query releases M walkers that hop by content and return home with probability $1-\alpha$. Their average readout estimates the restart resolvent of Proposition 4, a personalized PageRank (Page et al., 1999; Jeh & Widom, 2003); APPNP propagates by the same walk up to a degree rescaling (Gasteiger et al., 2019). Sampling keys from the softmax and averaging their values is Monte Carlo attention (Haris, 2025; Lee et al., 2026). Exact sampling still needs all N scores, so a walker saves gain, not compute, unless its proposal is sublinear, as with a maximum-inner-product index (Mussmann et al., 2017). *Tether:* the restart. *Budget:* gain plus $\ln M$, with a select readout only. *Null:* an absolute check at landing, or competitive killing by an absorbing sink. *Arrow:* timestamps compared at verification. *Writer:* at the encounter (untested).	Select buys a nat of $\ln N$ per nat of $\ln M$ when no sample outranks the target, and averaging buys nothing (Proposition 2); competitive killing fails as a sink does, and verifying walkers at fixed M keep abstention only while their routing mass above the threshold stays small, and inherit the gain wall (Proposition 3); the tether criterion (Proposition 4); timestamps cost no margin (Proposition 11(d)). Measured in Sections 3 and 4.	Proposition

Family	How it pays, and its writer	Failure predicted by	Label
Best-of-M verify-and-select*	Our construction: the walker's select readout as a layer of its own. It draws M candidates from a cheap proposal, scores each exactly, and keeps the best, or abstains if none passes. *Budget:* $\ln M$ plus the proposal's gain. *Null:* an absolute test. *Arrow:* a timestamp in the verification. *Writer:* the pairwise encounter of query and winner.	The select exchange (Proposition 2(a)) assumes that no sampled item outranks the target; when distractors can, more samples eventually hurt. A sampled absolute test (Proposition 3(c)); the evidence wall (Proposition 1).	Proposition
Langevin routing	Token states follow a noisy gradient flow on an energy set by content, and routing is drift toward attractors. A restoring force toward the input is the tether, and temperature against the energy's curvature sets the gain. Gradient-drift Langevin dynamics is reversible, so an arrow needs a drift that is not a gradient (Pavliotis, 2014). *Writer:* the energy's nonlinear terms.	Lock-in at low temperature, a failure of gain control (reading); circulation costs entropy at second order (Theorem 13; reading, since it is stated for finite Markov kernels).	Hypothesis
Spiking, event-driven routing	Spikes act as walkers, and winner-take-all circuits select (Maass, 2000); a neuron that does not fire abstains; spike timing carries the arrow; dendritic nonlinearities write at the synapse. The membrane leak is decay toward rest, a contraction, not a tether. Ours: spike-timing-dependent plasticity (Bi & Poo, 1998) learns the addresses, and on event-driven hardware (Davies et al., 2018) energy scales with the walkers, not with the pairs.	Selection among spikes (Proposition 2(a)); a firing threshold is an absolute test (Proposition 3(b)); timing is a hard arrow (Proposition 11(d)) whose storage Proposition 14(e) prices. All readings.	Hypothesis
Graph walker*	A greedy or beam walk on a proximity graph over the keys. *Budget:* hops times degree. Hierarchical navigable small-world graphs reach a logarithmic hop count under assumptions on the graph and on low-dimensional data (Malkov & Yashunin, 2020), and the earlier flat graphs need $\log^2 N$ hops (Malkov et al., 2014). Both searches stop on a relative rule and always return a result, so an absolute stopping threshold, which would make a null, is ours. The entry node is a starting point, not a tether. *Writer:* at the final node. Ours: the graph brings the walker's $O(N)$ proposal down to about $\log N$ hops, and it fails on keys with no neighbourhood structure.	The evidence wall (Proposition 1), lowered by search misses: neighbours are approximate, so recall falls below one before the wall. An absolute test on the returned score would abstain as Proposition 3(b) does.	Proposition
Hard, hierarchical selection			
Tournament and product-key tree	A query compares itself with a few keys at each level and keeps the best k branches, and a final hard or soft stage chooses among the survivors. In the two-level form, product keys, slot (i, j) has key $a_i \oplus b_j$, so each half of the query searches $\sqrt{N}$ sub-keys. With additive scores the exact top k lies in the product of the two halves' top-k sets, the top 1 of each half gives the global argmax, and the cost is $O((\sqrt{N} + k^2)\, d)$ (Lample et al., 2019). Memory layers scale this to millions of keys (Berges et al., 2025), and the inverted multi-index splits distances additively in the same way (Babenko & Lempitsky, 2012). A tree whose nodes weight their children by subtree partition functions is exactly the flat softmax (Proposition 2(c)), so its advantage lives in its hard stage; hierarchical softmax, whose nodes carry independently learned gates, is a different model (Morin & Bengio, 2005; Mnih & Hinton, 2008). *Tether:* each token keeps its own state, a residual. *Budget:* hard selection at each level moves $\ln(N/k^2)$ onto the query's evidence, and the soft final stage needs about $\ln k^2$ of gain. *Null:* a key check on the best candidate, or a sink over the shortlist. *Separation:* comparators against leaves, with factorized addresses. *Gain control:* hard decisions are discontinuous rather than of high gain. *Arrow:* bracket or insertion order, or timestamps. *Writer:* at the leaf and in the summaries of internal nodes.	The evidence wall (Proposition 1); a soft $k \times k$ final stage needs $2\ln((1+\sqrt{2})(k-1))$ nats of separation (Proposition 2(c)); clean factorized addresses are immune to cargo (Proposition 7); the key check abstains absolutely and the shortlist sink competes (Proposition 3); without a tether it coalesces, and a token one step from a two-cycle keeps its identity only for $\alpha < 0.618$ (Proposition 4(b)). Measured in Sections 3 to 5 and 7.	Proposition

Family	How it pays, and its writer	Failure predicted by	Label
Block-routed attention⋆	Block summaries are scored, the top k blocks are kept, a hard choice, and softmax runs within them (Yuan et al., 2025; Lu et al., 2025); the Routing Transformer instead assigns tokens to learned clusters (Roy et al., 2021). *Budget:* the evidence for the hard choice among the blocks, plus gain within the kept ones. *Null:* a null inside the soft stage competes only with the kept blocks. Native sparse attention always keeps the initial block, a sink, and MoBA always routes the current block, a tether. *Writer:* the destination MLP. Ours: it is robust to length with no temperature schedule while the block summaries carry the evidence.	Summaries dilute evidence, so it fails when the evidence is spread across blocks (Proposition 1); the gain wall within the kept set (Proposition 2); a competitive null over the bounded shortlist drifts only slowly (Proposition 3(a)).	Proposition
Indexer–reranker cascade	A cheap indexer selects the top k keys, 2,048 in DeepSeek-V3.2, and exact attention runs on them (DeepSeek-AI, 2025b;a); the indexer itself is still quadratic in length. *Budget:* $\ln(N/k)$ of the indexer's evidence plus $\ln k$ of gain. *Writer:* the destination MLP. Ours: the indexer's recall sets an evidence wall that the reranker cannot move; the source does not test this.	The indexer has its own code limit, set by its key dimension and match quality, and the reranker cannot recover its misses (Proposition 1 and Proposition 14(b)); the gain wall within k (Proposition 2).	Proposition
Mixture-of-experts routing⋆	The lattice applies to the router, which sends each token to k of E experts (Shazeer et al., 2017; Fedus et al., 2022). *Tether:* the residual, or a shared expert that every token uses (Dai et al., 2024). *Budget:* the top-k margin over E. *Null:* zero-computation or null experts, which compete inside the top k (Jin et al., 2025; Zeng et al., 2024). *Conservation on two sides:* load balancing; a linear assignment (Lewis et al., 2021); expert choice, which inverts the competition (Zhou et al., 2022). The router's z-loss is motivated by numerical stability and penalizes the level of the logits, not their spread (Zoph et al., 2022), so reading it as gain control needs an argument. Collapse onto a few experts is self-reinforcing, through training, not within a forward pass (Shazeer et al., 2017). *Writer:* the experts, so the router routes to writers.	The gain wall and the logit range over E experts, small and fixed (Proposition 2 and Proposition 14(c)); null experts compete over E, so they need no length schedule (Proposition 3(a)); collapse as a runaway is a reading, outside the class of Proposition 5.	Proposition
Mixture of depths, early exit	Routing over computation: a token may skip a block, and skipping is abstention. Mixture of depths trains with a capacity-limited top k over the sequence, which fixes the fraction skipped, and samples with a per-token absolute test adopted for causality (Raposo et al., 2024). Adaptive computation time halts on a per-token threshold (Graves, 2016), and confident adaptive language modelling lets its exit threshold depend on position by design (Schuster et al., 2022). *Writer:* the block used or skipped. Ours: without an absolute test, skip rates drift with sequence length.	A top k over the sequence is a rank decision, a fixed fraction and not causal; a per-token threshold is an absolute test (Proposition 3(b); reading).	Reading
Sorting and ranking networks	Routing by rank, made differentiable (Petersen et al., 2021; Cuturi et al., 2019; Mena et al., 2018; Adams & Zemel, 2011), as in the block sorting of sparse Sinkhorn attention (Tay et al., 2020). *Budget:* comparison depth, $\log^2 N$ for Batcher's practical networks (Batcher, 1968); depth $O(\log N)$ is possible, with impractical constants (Ajtai et al., 1983). *Arrow:* the rank order. *Writer:* after sorting. Ours: a tournament with a fixed schedule, robust but spending its depth even on easy inputs.	Each comparison is capped by its evidence (Proposition 1), and relaxed comparators compound over depth (reading).	Reading

Family	How it pays, and its writer	Failure predicted by	Label
Addressing and memory machines			
Address book	Content is hashed or quantized to a discrete address; a read fetches the slot, a write stores into it, and a key check verifies the stored key. Relatives include hashed attention (Kitaev et al., 2020), nearest-neighbour memories (Khandelwal et al., 2020; Wu et al., 2022), hash embeddings (Svenstrup et al., 2017) and Bloom filters (Bloom, 1970). *Tether:* write-through, a lookup added to the token's own value, not substituted for it. *Budget:* about $\log_2 N$ address bits keep buckets small, and selection within a bucket pays the rest. *Null:* an empty bucket or a failed key check, both absolute. *Separation:* index against payload. *Arrow:* overwrite on write, or timestamps. *Writer:* the update on write, plus a read MLP. With noisy keys the probes of locality-sensitive hashing grow like N^ρ (Indyk & Motwani, 1998; Har-Peled et al., 2012), but the code length per key stays logarithmic, at the rate of the binary symmetric channel (Section 3).	Identification's address bits (Proposition 14(a)); absolute nulls (Proposition 3(b)); timestamps cost $N \ln N$ nats of storage, and overwrite loses the history (Proposition 14(e) and Proposition 11(d)); the memory ledger (Proposition 14(d)). Addressing is measured in Section 3, and overwrite succeeds by construction (Section 8).	Proposition
Locality-sensitive-hashing attention	Random-rotation hashing, then attention within buckets (Kitaev et al., 2020). Several hash rounds raise recall: a model trained with one round scores 77.9% evaluated with one round and 99.9% with eight. The Reformer has no empty-bucket null: with shared queries and keys a query's own key is always in its bucket, and the query falls back to attending to itself. *Writer:* the destination MLP. With noisy keys the probes grow polynomially, like N^ρ (Indyk & Motwani, 1998; Har-Peled et al., 2012).	Hash misses lose evidence (Proposition 1); the softmax within a bucket has a gain wall (Proposition 2); a bucket check, if added, would be an absolute null (Proposition 3(b)).	Proposition
Vector-quantized memory*	Learned discrete codes, so the codebook is the address space (van den Oord et al., 2017). *Budget:* code bits. *Null:* an unused code, which acts as a null only with a defined null value. *Separation:* by construction. *Writer:* the code embeddings, plus the update on write. Codebook resets answer collapse or low usage (Dhariwal et al., 2020; Zeghidour et al., 2022; Łańcucki et al., 2020); the commitment loss is motivated by keeping encoder outputs from growing, not by collapse (van den Oord et al., 2017; Dhariwal et al., 2020). Ours: codebook collapse is the address book's runaway.	Identification's address bits (Proposition 14(a)); collapse feeds back through training, outside the class of Proposition 5 (reading).	Proposition
Bloom filter as a null	A membership test on hashed bits: an absolute test with one-sided errors, whose false-positive rate with k hashes into m bits is $(1 - e^{-kN/m})^k$ (Bloom, 1970; Broder & Mitzenmacher, 2004); a fixed rate needs m to grow with N. Ours: with its bits shared with retrieval, it gives any router a null calibrated for length, if the bits grow with N.	The membership ledger (Proposition 14(d)): about $N \ln(1/\varepsilon)$ nats; a Bloom filter uses about 1.44 times that.	Proposition
Holographic memory	Items are bound by circular convolution and superposed in one vector (Plate, 1995; Kanerva, 2009). *Budget:* the dimension d: crosstalk grows like $\sqrt{N/d}$, and the state has a fixed size, as in linear attention. A clean-up memory can answer "not close enough", a null. *Writer:* binding, a multiplicative writer, and the clean-up memory. Ours: capacity $N \lesssim d/(2 \ln V)$ for a vocabulary of V, a zero-margin estimate whose scaling agrees with Plate's fits but whose constants do not.	The memory ledger (Proposition 14(d)): d numbers hold N items of $\ln V$ nats. Our estimate gives about half a nat per number; outer-product states read by signs measure close to $1/\pi$ nat per number (Section 3).	Proposition

Family	How it pays, and its writer	Failure predicted by	Label
Modern Hopfield memory*	Retrieval descends an energy, and one update is softmax attention over the stored patterns X, $\xi \mapsto X\,\mathrm{softmax}(\beta X^\top \xi)$ (Ramsauer et al., 2021); it descends from Hopfield's networks (Hopfield, 1982). *Budget:* the gain β. Capacity grows exponentially with the dimension (Demircigil et al., 2017; Ramsauer et al., 2021), while the dense associative memory of Krotov & Hopfield (2016) has polynomial capacity. Separating one pattern takes a logit margin of about $2 \ln N$, a sufficient condition (Ramsauer et al., 2021). *Gain control:* the update has three kinds of fixed point, a global average, metastable mixtures and single patterns; the global fixed point is unique at small β, and β sets how many patterns are averaged. *Tether:* none. The update replaces the query with a mixture of stored patterns, so iteration forgets the query: the query is only a starting point, and re-injecting it at every step would be a tether. *Writer:* the energy's separation function.	Exponential capacity is the key channel's code limit, $\ln N_{\mathrm{ev}} \approx (d-1)C(c)$ (Proposition 1 and Proposition 14(b)); the gain wall in β (Proposition 2); without a tether, iteration collapses queries onto stored patterns (Proposition 4; reading). Re-injecting the query at every step with weight $1-\alpha$ is the tether, and Proposition 18 gives its certificate: retrieval keeps a unique fixed point, and the query's identity, whenever $\alpha\beta D^2/4 \le 1$ for patterns of diameter D, and the constant is sharp.	Proposition
External nearest-neighbour memory	A hard retrieval stage followed by a soft reading (Wu et al., 2022; Khandelwal et al., 2020; Lewis et al., 2020; Borgeaud et al., 2022; Guu et al., 2020). None of these systems uses a distance threshold: Memorizing Transformers gate with a constant per head, the nearest-neighbour language model mixes with a fixed weight, and RAG and RETRO always use the top k. REALM's null document is a competitive null within a retrieval cascade. Approximate search recalls about 90% of the true neighbours (Wu et al., 2022). *Writer:* the reader network. Ours: an absolute distance threshold as the null, and scaling with length that needs no gain, limited by the index's recall.	The index's recall is the evidence wall (Proposition 1); a distance threshold would be an absolute null (Proposition 3(b)), and a null document is a competitive one (Proposition 3(a)).	Proposition
Content-addressable memory	Ternary content-addressable memory compares the query with every stored word, each on its own match line, in one cycle (Pagiamtzis & Sheikholeslami, 2006): an address book in silicon. Each match line is an absolute test, and a missing hit is an explicit null; with don't-care bits a match is exact only on the specified bits.	A word needs at least $\log_2 N$ bits to address N items uniquely (Proposition 14(a)); match lines abstain absolutely (Proposition 3(b)).	Reading

Family	How it pays, and its writer	Failure predicted by	Label
Physical and analog machines			
Relaxation network	Tokens are nodes joined by conductances, and each node leaks to its clamped input. With reciprocal conductances and leaks in proportion, Kirchhoff's laws give the tethered router's fixed point exactly (Proposition 20; Doyle & Snell, 1984). Nonlinear devices at the nodes make it a deep equilibrium (Bai et al., 2019), and Hopfield's analog circuits are close relatives (Hopfield, 1984). *Tether:* the leak to the clamped input. *Budget:* conductance contrast, capped by the devices' dynamic range. A contrast of 10^3 to 10^4 is the best case of two states in an isolated device; integrated crossbars use one to two decades of conductance, and analog devices report on/off ratios of 2 to 100 (Le Gallo & Sebastian, 2020; Prezioso et al., 2015; Ielmini & Wong, 2018; Joshi et al., 2020; Zhao et al., 2020). *Null:* a conductance to ground, competitive unless it scales with the total conductance, or a threshold at the output, which is absolute. *Separation:* couplings against potentials. *Gain control:* adaptive conductances need loop gain below the leak. *Arrow:* non-reciprocal elements, which cost power. *Writer:* nonlinear devices. Analog crossbars solve such linear systems in one step per right-hand side, with operational amplifiers in feedback, errors of 3–10% and matrices that are reused rather than set from content on each input (Sun et al., 2019); nonlinear resistive networks train by equilibrium propagation, simulated at circuit level (Scellier & Bengio, 2017; Kendall et al., 2020). Its digital form, a truncated restart resolvent, trains end to end in Section 10. That a physical network routes by content is the hypothesis.	For ideal circuits: the tethered resolvent (Proposition 20) and its criterion (Proposition 4); the dynamic-range wall of one competitive stage, where a majority needs $N - 1 \leq G_{\max}/G_{\min}$ with the leak as the null weight, which puts the wall near $N \approx 10$ to 100, or lower for the weakest devices (Proposition 19 and Proposition 14(c)); nulls by their test (Proposition 3); circulation costs entropy at second order (Theorem 13); a linear circuit cannot write (Proposition 10). Adaptive conductances are a reading of gain control.	Hypothesis
Optical correlator	A matched filter compares a scene with a stored reference at every shift (VanderLugt, 1964; Weaver & Goodman, 1966), not arbitrary query–key pairs. *Budget:* optical contrast, capped by the detector's dynamic range. *Null:* a reference beam, which competes. *Arrow:* non-reciprocal optics, whether magneto-optic, nonlinear or time-varying (Jalas et al., 2013); reciprocity constrains the physical scattering matrix, not computed scores, so an asymmetric encoding needs no isolator. *Writer:* weak optical nonlinearities, or electronics after detection. Ours: contrast sets a hard gain wall, so large N needs a hard or hierarchical stage.	Contrast as the logit window, if intensities are normalized (Proposition 19); a competitive reference beam (Proposition 3(a)); linear optics cannot write (Proposition 10). All readings.	Hypothesis
Chemical, biological and social machines			
Adaptive immune repertoire	Clonal selection is hard selection followed by amplification (Burnet, 1957; 1960; Hodgkin et al., 2007), graded by affinity. Negative selection removes self-reactive clones, an abstention against self, though thymic deletion is imperfect (Klein et al., 2014); regulatory T cells restrain responses, a gain control, and autoimmunity can also follow their failure (Sakaguchi et al., 2008); affinity maturation learns the addresses (Victora & Nussenzweig, 2022). *Writer:* memory and effector cells. Ours: autoimmunity is a failed abstention test, and a cytokine storm is a runaway, which the clinical literature describes as failed negative feedback, with a positive loop only suggested (Fajgenbaum & June, 2020).	Abstention against self (Proposition 3) and runaway (gain control). Both readings.	Hypothesis

Family	How it pays, and its writer	Failure predicted by	Label
Gene regulatory networks	Transcription factors bind promoters competitively or cooperatively, and the Hill coefficient, the log–log sensitivity of the bound-to-free odds, is a gain (Hill, 1910; Bintu et al., 2005); basal expression sets the reference level. Occupancy is graded, and its steepness comes from cooperativity, so a promoter is not an absolute threshold.	Cooperative feedback can be multistable (gain control); a true threshold would be an absolute test (Proposition 3(b)). Both readings.	Hypothesis
Markets and auctions	In logit choice, utilities play the role of logits and the outside option sits in the normalizer, a competitive null (McFadden, 1974); in proportional allocation a share is linear in the bid, so the logit is log(bid) (Kelly, 1997; Johari & Tsitsiklis, 2004). A reserve price compared with the highest bid is an absolute test that does not depend on the number of bidders (Myerson, 1981). Two-sided logit matching is entropic optimal transport, solved by iterative proportional fitting, which is Sinkhorn's algorithm, with singles as dustbins (Choo & Siow, 2006; Galichon & Salanié, 2022; Peyré & Cuturi, 2019). The retained endowment is the tether. *Writer:* production.	A reserve price is an absolute test (Proposition 3(b)); clearing and outside options compete (Proposition 3(a)); the two-sided law on each side. All readings.	Analogy
Consensus protocols, an anti-example	Gossip and voting protocols want collapse, so they drop the tether: consensus is the designed outcome (DeGroot, 1974; Boyd et al., 2006; Olfati-Saber et al., 2007). Friedkin and Johnsen's anchored model, $Y_t = \alpha W Y_{t-1} + (1-\alpha)Y_1$, is the tethered iteration of Proposition 4 in social form, with DeGroot's model as its limit $\alpha \to 1$ (Friedkin & Johnsen, 1990); fixed leaders also block consensus (Olfati-Saber et al., 2007).	Without a tether, consensus by design (Proposition 4).	Analogy
Packet-switched networks	Routing tables are an address book, and header against payload is separation. The time-to-live field bounds a packet's travel: RFC 791 defines it in seconds (Postel, 1981), RFC 1812 has routers decrement it once per hop (Baker, 1995), and IPv6 names it the hop limit (Deering & Hinden, 2017); it guards against loops, and the network's null for an absent destination is "no route". Sequence numbers are the arrow (Eddy, 2022). Congestion control by additive increase and multiplicative decrease is gain control (Jacobson, 1988; Chiu & Jain, 1989); it oscillates about the efficient, fair point rather than settling at a fixed point.	A hop limit as a bounded search, like Proposition 3(c); sequence numbers as a hard arrow (Proposition 11(d)), with storage that Proposition 14(e) prices; congestion control as gain control. All readings.	Reading
Recurrent machines with bounded state			
Gated linear attention and state-space models	A recurrence keeps a state of fixed size: Mamba, RWKV, RetNet and gated linear attention (Gu & Dao, 2024; Peng et al., 2023; Sun et al., 2023; Yang et al., 2024b). Decay gates contract the state, which bounds it but is not a tether; Mamba's and gated linear attention's decays depend on the input, while RWKV's and RetNet's are learned constants. *Null:* an output gate. A normalizing denominator, as in linear attention (Katharopoulos et al., 2020), averages and cannot route nowhere, and gated linear attention and RetNet have none. *Arrow:* the recurrence itself. *Writer:* the MLP and selective, gated writes; Mamba folds the MLP into its block. Recall is capped by the state: any causal recurrent model needs $\Omega(N)$ bits of state to solve multi-query associative recall (Arora et al., 2024), and copying is limited by the state's bits (Jelassi et al., 2024). Robustness to length is mixed. It holds for decayed models in perplexity (Yang et al., 2024b) and for some recall tasks (Gu & Dao, 2024), but Mamba's length generalization is limited (Ben-Kish et al., 2025), and so is that of the decay-free delta rule (Yang et al., 2024c).	The memory wall of a bounded state (Proposition 14(d) and Corollary 15(iii)), measured for outer-product states in Section 3; the state is bounded only with contracting writes (Bruce, 2026d); a denominator sums over all N, as in the law of sums (reading), while an output gate tests each token.	Proposition

Family	How it pays, and its writer	Failure predicted by	Label
The delta rule	The write $S \mapsto S\big((1-\eta\lambda)I - \eta kk^\top\big) + \eta vk^\top$ replaces the value stored at a key (Schlag et al., 2021; Yang et al., 2024c): the error-correcting update of Widrow & Hoff (1960). *Arrow:* overwrite gives latest-write semantics without spending positional margin. It is exact when the write strength is one and the keys are unit-norm and mutually orthogonal, and it loses the history: it answers latest-value queries only. *Separation:* key against value. *Gain control:* for $\eta, \lambda \geq 0$ the write is non-expansive if and only if $\eta(\lambda + \|k\|^2) \leq 2$; at the endpoint $\eta\|k\|^2 = 2$ with $\lambda = 0$ it is a Householder reflection (Grazzi et al., 2025). Non-expansiveness bounds the difference between two trajectories driven by the same inputs, not the state: with $\lambda = 0$ a non-expansive write can grow without bound (with $\eta = 1$, scalar keys $k_t = 1/t$ and unit values, the state grows like $\log t$). The state is bounded only under strict contraction, which needs decay: with keys of norm at most K, values of norm at most V and $q = \max(|1-\eta\lambda|, |1-\eta(\lambda+K^2)|) < 1$, $\|S_t\| \leq q^t\|S_0\| + \eta VK(1-q^t)/(1-q)$ (Bruce, 2026d). DeltaNet as deployed, with write strength in $(0,1)$, unit keys and no decay, is non-expansive but not strictly contracting, and its length generalization is limited (Yang et al., 2024c); Gated DeltaNet's decay gate supplies the contraction (Yang et al., 2025). *Writer:* the delta update itself.	The memory ledger (Proposition 14(d)): in the memory test of Section 3 the online delta rule saturates and recalls the newest items; overwrite is a hard arrow that answers latest-value queries only (Proposition 11(d) and Table 2); a bound on the state needs strict contraction.	Proposition
Test-time-training layers	The state is a small model updated by gradient steps, so writer and memory merge (Sun et al., 2025); the inner learning rate is a gate that depends on the input. With a linear inner model and a squared loss, one inner step is a delta-rule write. Reading the inner rate as gain control is ours; the source gives no stability bound.	The memory ledger of a bounded state (Proposition 14(d)); the delta rule's measured memory (Section 3).	Proposition
Neural cellular automata	Each cell sees only its 3×3 neighbourhood and updates its own state through a residual (Mordvintsev et al., 2020), so the budget is paid in steps: information moves a bounded distance per step. *Writer:* the per-cell update. Ours: long-range recall needs many steps.	No proposition prices locality; the binding limit is a light cone.	Reading
Mixers that do not depend on content			
Wavelet and multiresolution mixers	A hierarchy that ignores content: a fixed tree of sums and differences (Mallat, 1989), not of maxima. In WaveMix a fixed wavelet transform outperforms 2×2 max pooling (Jeevan & Sethi, 2022; Jeevan et al., 2024), a measured comparison of sums with maxima.	None: a frozen operator routes nothing by content, so it lies outside the propositions.	Reading
Graph and geometric machines			
Message passing on learned nearest-neighbour graphs	Messages $\phi(x_i, x_j)$ are written on the edges of a sparse graph, and a self-loop or residual keeps each node's own state (Gilmer et al., 2017; Battaglia et al., 2016; 2018). Only the dynamic graph CNN chooses its graph by content, through a dense $O(N^2)$ distance computation, and it aggregates by a maximum (Wang et al., 2019); the others route on given graphs. Ours: the clearest existing case of a writer that sees both ends of an edge, made affordable because routing is sparse.	Neighbours are chosen by feature distance, so the evidence wall applies (Proposition 1); routing each node to $k > 1$ neighbours is not a functional graph, so the cycle analysis of Proposition 4(b) needs extending.	Proposition
Hyperbolic and tree embeddings	Hierarchy as geometry: trees embed in hyperbolic space with low distortion, and greedy routing there is guaranteed to arrive (Nickel & Kiela, 2017; Sarkar, 2011; Kleinberg, 2007). The number of hops equals the path length in the tree, so it is logarithmic only for balanced hierarchies, and the bits of precision must grow with the longest path (Sala et al., 2018). Hyperbolic attention is dense, not tree routing (Gulcehre et al., 2019).	Precision is the currency, with bits in proportion to path length; the evidence wall (Proposition 1).	Reading

Family	How it pays, and its writer	Failure predicted by	Label
Random-feature attention	Positive random features turn softmax attention into linear attention, whose causal form is a prefix-sum state (Choromanski et al., 2021; Peng et al., 2021; Katharopoulos et al., 2020): a machine with bounded state. A gated variant adds a decay that depends on the data (Peng et al., 2021), and the state's capacity is limited (Schlag et al., 2021).	The memory ledger (Proposition 14(d)): the causal form keeps a fixed state; outer-product states read by signs measure close to $1/\pi$ nat per number (Section 3).	Proposition
Composite machines			
Hybrids with a recall reserve	A bounded-state recurrence whose writes contract keeps a few attention layers as a recall reserve, whose cache stays linear in context (Bruce, 2026d). Pure recurrences lag on recall and in-context tasks, and the hybrids recover them (Lieber et al., 2024; Waleffe et al., 2024). Deployed hybrids keep attention in 1/8 of their layers (Jamba, MiniMax-01) to 1/4 (Qwen3-Next, and Kimi Linear's nominal 3:1) (Lieber et al., 2024; MiniMax, 2025; Qwen Team, 2025; Kimi Team, 2025). Kimi Linear as released has attention in 7 of 27 layers, NVIDIA's 8B hybrid in 4 of its 28 sequence mixers (Waleffe et al., 2024), and Griffin in a third, with local attention only (De et al., 2024). These fractions are observations, not predictions: concave pricing alone predicts a fraction of 0, 1 or a binding cap, and the level set $\{1/2, 1/3, 1/4, 1/6, 1/8\}$ used to state predictions is a convention, not a consequence (Bruce, 2026d). Kimi Linear's own ablation is flat between 1/2 and 1/8, and MiniMax-01 reports that shallow models need more softmax layers. Bruce (2026d) registers a sweep of the fraction over $\{0, 1/8, 1/4, 1/2, 1\}$ to measure the trade-off; it has not run.	The memory ledger for the recurrent layers (Proposition 14(d)); the attention layers pay memory in full, with a cache linear in N; concave pricing predicts 0, 1 or a binding cap (Bruce, 2026d).	Proposition
Retrieval-augmented cascades	An external index retrieves the top k, a hard stage limited by evidence, and attention reads them, a soft stage limited by gain over a bounded set (Lewis et al., 2020; Borgeaud et al., 2022; Guu et al., 2020; Khandelwal et al., 2020; Wu et al., 2022).	The index's recall (Proposition 1); the gain wall within k (Proposition 2); a competitive null over the bounded set (Proposition 3(a)).	Proposition
Walker plus tree*	Our construction: a tree proposes candidates and walkers verify them, so the budget splits three ways, among the tree's evidence, samples and gain.	The tree's evidence (Proposition 1); samples (Proposition 2(a)); the soft stage (Proposition 2(c)).	Proposition