The Transformer Abides: What Attention Needs, and Why It Endures

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

Abstract

Nine years after its introduction, the transformer remains the default sequence model. The usual explanations are real: parallel matrix multiplication on accelerators, predictable returns to scale, a co-evolved ecosystem and exact recall from an uncompressed context. But they do not explain why most modifications barely move accuracy, which properties are essential, why the changes practice adopted took the forms they did, or where the design can still move. This paper is the entry point to a collection of papers that treat a token-mixing layer as an averaging operator rebuilt from its own input and iterated through depth. Such a layer faces six failure modes (collapse to consensus, contamination of content by routing, runaway feedback, the inability to abstain, dilution as context grows, and the inability to represent direction). Each has a protection, proved within a declared class, and a signature measured at small scale when that protection is removed; for gain control, which protects against runaway feedback, that signature is synthetic. We read softmax attention in a pre-normalized residual block as the one known point where all six protections are cheap at once under the prices of parallel hardware and gradient descent; this is an interpretation, not a measured optimum. Two mechanisms, measured at small scale, help explain why its neighbourhood is flat: trained networks compensate local deficits through spare capacity, and, in ten scored tests, making a protection explicit did not beat a block that already supplies it. The anchor, the residual or input path that protects against collapse to consensus, is the one protection whose loss compensation provably cannot repair: under strict mixing, where repeated averaging contracts token differences toward zero, no extra capacity of bounded gain (bounded sensitivity) brings back what is lost without it, and in small-model ablations nothing else in the block made up for it. This is a proved bound; a trained test that adds capacity is pending. Many changes adopted in the mixing layer and its block since 2017, among them pre-normalization, QK-norm, sinks and output gates, read as re-supplies of the same protections; the account is silent on others, such as changes to the MLP. The room that remains for the design to move is where a protection is patched (met only by a fix that works over a limited range, such as the trained length) or paid at a price that another design could avoid. It includes abstention and retrieval at long context, governed by a two-sided log-N law in the context length (with bounded scores, no fixed rule for abstaining keeps both as the context grows), and bounded memory, whose price for exact recall is a per-head capacity. It also includes decisions that sum over all competitors where reading the largest would do, and the writer of new features (in a transformer, the MLP and the output map), for which the collection has no theory yet of what a good writer computes. Along the way the collection establishes results that stand on their own: attention has an arrow that default statistics of a single operator cannot see; depth is a distribution over path lengths; converged adaptive averaging can amplify changes in its input and overturn correct predictions; and softmax obeys length laws that decisions on maxima escape while the competing scores stay below the decision's threshold. We state what is proved, what is measured, what is interpretation, and what is still pending.

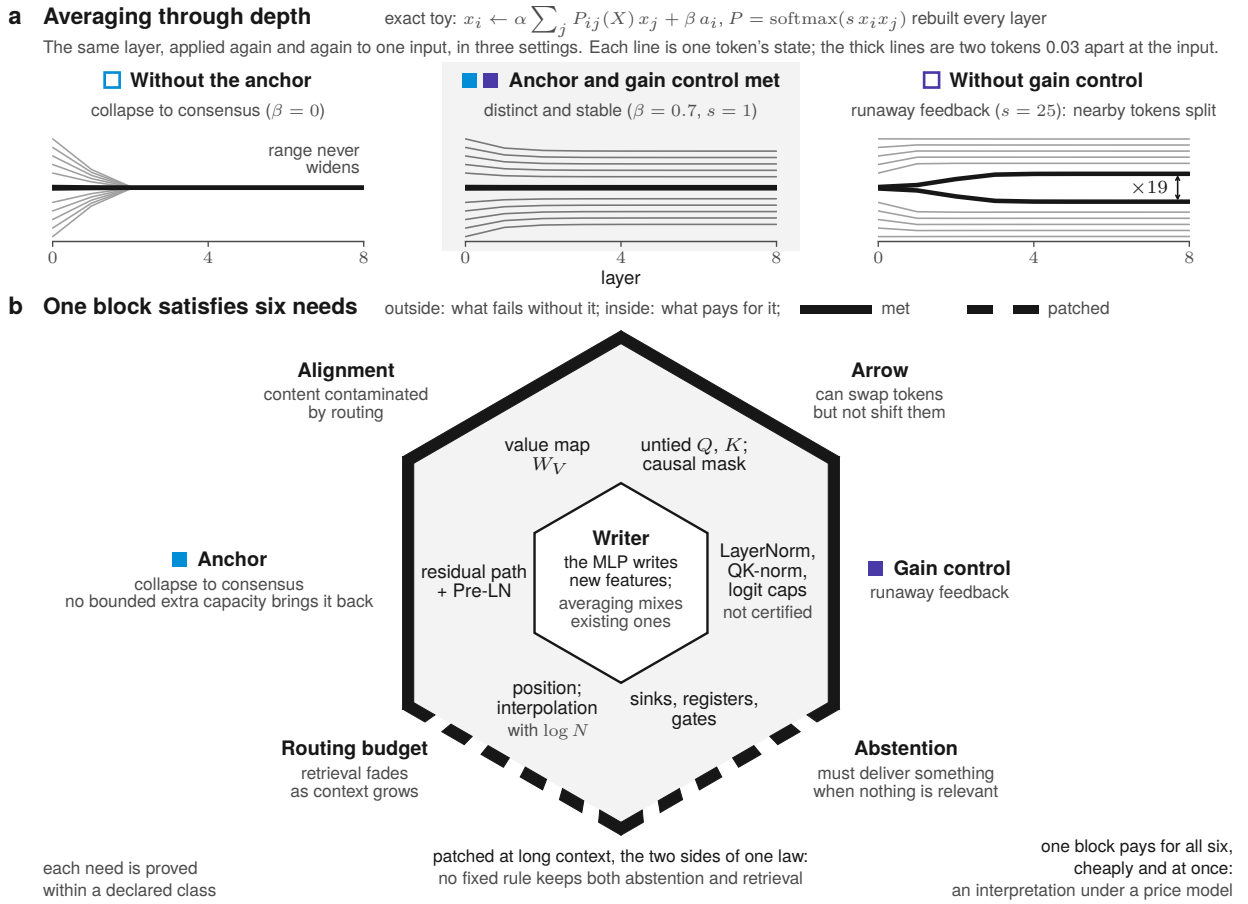


Figure 1: **Averaging iterated through depth can merge or split its tokens unless its needs are met (a), and the pre-normalized block pays for all six needs (b). (a)** An exact toy: the anchored iteration of M§1.4 (the panel's formula, with $\alpha = 1 - \beta$), applied eight times to twelve one-dimensional inputs a_i. In the middle, the anchor ($\beta = 0.7$) and gain control are met (stability margin $m = \beta - 2\alpha s R^2 = 0.1 > 0$ at logit scale $s = 1$ and input radius $R = \max_i |a_i| = 1$): the iteration has one attracting state, and the two thick tokens, 0.03 apart at the input, end 0.02 apart. Each side changes one of these settings. Without the anchor ($\beta = 0$ at $s = 1$, which also makes $m < 0$), every output is a convex combination of the previous states, and the tokens collapse to consensus (rank collapse; Dong et al., 2021). Without gain control ($s = 25$ at $\beta = 0.7$), the same input has four attracting states; in the one it reaches, the two thick tokens end 0.57 apart, so a small difference in the input moves the output nineteen times as far. The margin is sufficient, not necessary: this toy keeps a single attracting state up to $s = 8$. **(b)** The six needs are the conditions of Section 3.2, each proved within a declared class (Table 2); outside each side is what fails without the need, inside is what in the block pays for it, the writer at the centre lies outside the six, and blue and violet mark the two needs drawn in (a). The dashed sides at the bottom are patched rather than met at long context, where the design can still move (Section 3.7): with bounded logits and a null option (to deliver nothing) that absent and present queries share, no fixed rule keeps both abstention and retrieval (M§9.1). On the right, gain control is met by norm control, which is not a certificate (M§7.5). On the left, under strict mixing, no extra capacity of bounded gain brings back what is lost without the anchor (M§5.5). The toy is an illustration, not a measurement of a trained model: its anchor re-injects the input and keeps a floor, whereas at initialization the block's residual path with Pre-LN slows the loss of distinctions to a polynomial in depth and sets no floor (M§2.6). That the block pays for all six cheaply and at once is an interpretation under a price model, not a measurement of prices.

1 The question

Nine years after it was introduced (Vaswani et al., 2017), the transformer is still the default model for sequences.[1] It has not lacked challengers, and several have beaten it on one axis: linear attention and state-space models on the cost of inference (Katharopoulos et al., 2020; Gu et al., 2022; Gu & Dao, 2024), recurrent memories on the memory cost of long contexts (Schlag et al., 2021; Yang et al., 2024; 2025), and sparse attention on long-context compute (Yuan et al., 2025; Lu et al., 2025). None has replaced it. Practice instead kept the design and changed it in small, specific ways. Normalization moved in front of each sublayer (Wang et al., 2019; Xiong et al., 2020); queries and keys were normalized and logits capped (Henry et al., 2020; Gemma Team, 2024); sink tokens, registers, learned sink logits and output gates were added (Xiao et al., 2024; Darcet et al., 2024; OpenAI, 2025; Qiu et al., 2025); logits were rescaled with context length (Chiang & Cholak, 2022; Peng et al., 2024); hybrids kept a fixed fraction of attention layers (Lieber et al., 2024; Qwen Team, 2025; Kimi Team, 2025); the cache was compressed into a latent representation (DeepSeek-AI, 2024); and looped models re-inject their input at every step (Bansal et al., 2022; Geiping et al., 2025).

The program behind this paper began from four questions: how did we land on transformers; why do they seem to work so well for so many things; why have they been a fixed point, or a strong attractor, in the space of architectures; and why have we not moved on. We ask them as two. *Why transformers?* asks about the design's origin and its generality. *Why are we still here?* asks about its stability, and about where it can still move. We do not claim to answer either fully. We claim a partial answer, with a stated standard of evidence, and we say which parts of it are proved, which are measured, which are interpretation and which are still pending.

What the usual answers leave open. The familiar explanations are real, and Section 2 presents them with their best evidence. The transformer turned sequence modelling into large matrix multiplications, the operation accelerators improved fastest; scaling laws made capability predictable and purchasable; kernels, serving systems, recipes and people co-evolved around it; attention keeps every past token and so supports exact recall; and the block trains. These explain why this lineage was found first and why it was funded. They leave four things open. Why do most published modifications barely move accuracy: why is the design's neighbourhood flat? Which of its properties are essential, and why exactly those? Why did the changes that practice adopted take the forms they did? And where can the design still move, and what must a successor supply?

Our answer. The collection treats a token-mixing layer as an averaging operator whose weights are rebuilt from the layer's own input, applied again and again through depth. Such a layer faces six failure modes: collapse toward a consensus, as every token drifts toward a common average; contamination of content by routing, when what decides where to look is mixed into what is carried; runaway feedback, since the weights are computed from the tokens they update; the inability to abstain, to deliver nothing when nothing is relevant; dilution as context grows, as more tokens compete for the same attention; and the inability to represent direction, to move information one way along a sequence. Each has a protection proved within a declared class of models, and a removal signature, measured at small scale: the failure that appears when that protection is taken away. For gain control, which protects against runaway feedback, the measurement is synthetic. A seventh property, writing new features, lies outside the six failure modes and their protections (Section 3.2, Table 2). We read softmax attention in a pre-normalized residual block as the one known point where all six protections are cheap at once under the prices of parallel hardware and gradient descent. That reading is an interpretation under a price model, at prices that co-evolved with the design, not a measured optimum (Section 3.3). Hardware sets the prices; the protections, as conditions on the block, are what is being priced.

Two mechanisms, measured at small scale, help explain why the block's neighbourhood is flat: trained networks make up for a weakened component with spare capacity elsewhere (compensation), and turning a protection into an explicit, adjustable setting (a knob) does not beat a block that already supplies it (Section 3.4). The anchor, the residual or input path that protects against collapse toward a consensus, is

[1] A monograph that this paper opens gathers the program's results with their evidence (Bruce, 2026a). We cite its sections as M§9.2, meaning its Section 9.2, and give its chapters by number in the tables.

the one protection whose loss compensation provably cannot repair: under strict mixing, where repeated averaging contracts token differences toward zero, no extra capacity of bounded gain (bounded sensitivity) brings back what is lost without it, and in small-model ablations nothing else in the block made up for it. That is proved; a trained test that adds capacity is pending. Why would designs jump between discrete choices rather than drift smoothly? If design prices are concave, so that a blend of two allocations costs at least the average of their prices, a cheapest way to provide a fixed total of protection sits at a corner of the feasible set, where every protection but at most one is either off or at its cap; designs would then jump between corners rather than drift. The premise that design prices are concave is unmeasured, so this step is a hypothesis (Section 3.5).

Read retrospectively, many changes adopted in the mixing layer and its block since 2017 re-supply the same protections, and most of these read as responses to the failure each one fixes; the account is silent on others, such as changes to the MLP (Section 3.6).

The room that remains is where a protection is patched (met only by a fix that works over a limited range, such as the trained length, not by the block's structure at every length) or paid at a price that another design could avoid: abstention and retrieval at long context, governed by a two-sided log-N law in the context length (with bounded scores, no fixed rule for abstaining keeps both as the context grows); bounded memory, where a fixed-size state caps recall at a per-head capacity; softmax decisions that sum over all competitors where reading the largest score would do; and the writer of new features, for which the collection has no theory yet (Section 3.7). The main open problem the framework leaves is how much of each protection's price the MLP, which writes, can absorb, and at what width (M§§11.7, 19.6).

What the question uncovered. Answering the question turned up results that stand on their own, for readers who do not care why transformers persist (Section 4). Attention has an arrow: a direction carried by circulation, the net one-way flow of attention around loops of tokens. The default statistics of a single operator, such as how strongly it pulls tokens toward consensus, cannot see this direction. Depth is a distribution over path lengths, the number of averaging steps a signal takes through the layers, and short paths decide what survives. Adaptive averaging can settle to a fixed point and still amplify changes in its input, so settling is not solving. Softmax obeys length laws that decisions on maxima escape while the competing scores stay below the decision's threshold. And the transformer already contains protections that proposals add explicitly.

What holds in trained networks. Many of the collection's results are proved for an idealized object: one operator applied over and over, a background drawn independently or replaced by its mean, keys constructed by hand. A trained network applies a different, input-dependent operator at every layer, with residual paths and MLPs between. Section 5 says, claim by claim, which results hold exactly for every network of their class, which hold under premises that must be checked network by network, which are shown in an idealized setting, and which are measured in trained networks, at what scale.

What this paper does and does not do. It is the entry point to a collection of papers and the opening of the monograph that gathers them (Bruce, 2026a). Its claims are made and evidenced in the monograph's chapters, which it cites by section for every result except two remarks on graph-consistency regularization, marked as unreleased work. For three results, the weight-decay cap on the logit scale, the divergence detector's confound and the row-by-row record of the knob study, the monograph states the result and defers its full record to a later version. It re-proves nothing and reports no new experiment. Each of the collection's claims carries one of five labels (proved, measured, partial, pending, interpretation), and the usual answers this paper reviews carry a sixth, conventional (Appendix B). The decisive tests that are still pending are named, with what each would change. It does not propose a better architecture. The account expects successors to win mainly on resource axes, such as memory, length and speed, at matched quality (Section 6).

Contributions.

- A fair account of the usual answers, with what the collection adds to each: support, a mechanism, a refinement, a correction, an adjacent result, or nothing (Section 2, Table 1).

- The conditions account of why the transformer persists, step by step, each step with its strongest evidence and its scope (Section 3).
- Where the design can still move, and what any successor must still supply (Section 3.7).
- A guide to what the program found beyond the question, by theme (Section 4, Table 3).
- A statement of what transfers: which results hold exactly for trained networks, which under premises, which in idealized settings, and which are measured, with the models and the scale (Section 5, Table 4).

Appendix A introduces the monograph and suggests reading paths through it, Appendix B sets out the evidence standards, Appendix B.4 lists the tests still pending, and Appendix C lists every claim with its label and its source.

2 What the usual answers explain

The question of why the transformer has lasted is old, and the field has good answers to it. They fall into three families: why this lineage won, which properties of the design matter, and how the field behaves around an incumbent. We give each its best case, because the account in Section 3 is meant to complement them, not to replace them. Two recent papers take stock of the transformer from other sides. One considers the challengers' perspective: *The End of Transformers? On Challenging Attention and the Rise of Sub-Quadratic Architectures* (Fichtl et al., 2025). The other considers its failure modes in safety-critical use: *In Transformer We Trust? A Perspective on Transformer Architecture Failure Modes* (Mondal & Jagtap, 2026).

2.1 Why this lineage won: hardware, scale and history

The transformer turned sequence modelling into large, dense matrix multiplications, the one operation that accelerators were improving fastest, and ideas tend to win when they suit the machines of their day (Hooker, 2021). Loss was then found to fall as a smooth power law in model size, data and compute (Kaplan et al., 2020; Hoffmann et al., 2022). This made capability predictable and purchasable: buying scale became a safer bet than inventing architectures, and research risk became capital expenditure. Kernels, serving systems, training recipes and people's intuitions co-evolved around the design (e.g., Dao et al., 2022), so a transformer's compute became cheaper and less risky than a challenger's. Proving an alternative is expensive in its own right: wins at small scale often fail to hold at large scale (Tay et al., 2023), and a frontier run is too costly to gamble on an unproven design.

History did its part. Attention entered machine translation to align a decoder with its source (Bahdanau et al., 2015). The 2017 contribution, a design found where translation data and accelerators met, was largely a subtraction: attention without recurrence or convolution (Vaswani et al., 2017). Next-token prediction then let modality after modality be cast as one problem. Benchmarks grew up around what the incumbent made possible, which can make it look better suited to the tasks than it is. And in-context learning, picking up a task from examples in the prompt (Olsson et al., 2022), turned one model into a platform, whose use funded the next round of scale.

2.2 Why this design: the properties that matter

Turning to the design's own properties, one common answer is that attention has a weak inductive bias, assuming almost nothing about which inputs interact. On this view, general methods that exploit computation eventually beat methods that build in what we know (Sutton, 2019), as vision transformers showed once data was plentiful (Dosovitskiy et al., 2021). A second is the training signal: with a causal mask, one parallel pass over a sequence yields a prediction loss at every position (Vaswani et al., 2017). A third is memory. Attention keeps every past token and looks up what it needs when it reads, while fixed-size memories provably cannot recall arbitrarily long inputs (Arora et al., 2024a;b; Jelassi et al., 2024), which is why efficient hybrids keep some attention layers. A fourth is the block's organization: each layer reads from a shared residual stream and writes an edit back (Elhage et al., 2021); attention moves information between tokens, and the MLP transforms each token and can be read as a key-value memory (Geva et al., 2021).

Then come trainability and the form of the operations. Much of the 2017 design is conditioning, through scaled dot products, normalization and residual paths. Later refinements of that conditioning, such as moving the normalization before each sublayer, were about optimization (Xiong et al., 2020). On this view architectures are selected by their loss landscapes as much as by what they can express. Two further answers point to the form of attention itself: even the input-dependent part of attention is produced and consumed by matrix multiplications, and softmax can concentrate almost all of its weight on one position, which exact copying and retrieval need (Jelassi et al., 2024). Finally, the architecture is not the algorithm: much of what would otherwise need a redesign can be learned inside the weights (e.g., Olsson et al., 2022), so the block diagram stays fixed while the computation changes.

2.3 Why it stays: how the field behaves

The transformer absorbs its challengers as components. Mixture-of-experts layers, state-space layers and sparse and linear attention ended up inside transformer stacks (e.g., Lieber et al., 2024; OpenAI, 2025; Qwen Team, 2025), and gating moved into attention itself (Qiu et al., 2025). Lineages built to replace it redesigned themselves for parallel training and converged towards attention (e.g., Gu & Dao, 2024), as the duality between state-space models and linear attention makes explicit (Dao & Gu, 2024). Weaknesses are worked around above the architecture, with chain of thought, retrieval and tools (e.g., Merrill & Sabharwal, 2024).

The design space around the transformer is flat: when dozens of published modifications were re-implemented in one codebase, most did not help (Narang et al., 2021). Scale sorts architectures, so far in the transformer's favour: alternatives that match it at small scale tend to fall behind as compute grows (Tay et al., 2023). And what has lasted may be a template rather than a block, since almost every component has changed since 2017 (e.g., Labonne, 2026; Xie et al., 2025).

Table 1 lists the twenty-one explanations above, what the collection says about each (its bearing), and where this paper argues for that assessment.

2.4 What they leave open

Between them the usual answers explain why this lineage was found first, why it was funded, and why recall keeps attention in efficient hybrids. They leave four questions open.

- **Why is the neighbourhood flat?** That most modifications do not transfer, and that small-scale wins often fade with scale, is observed (Narang et al., 2021; Tay et al., 2023); it is not explained.
- **Which properties are essential, and why exactly those?** Calling the residual stream "a bus" is an intuition without a criterion. Recall explains why hybrids keep some attention, and trainability why the residual stream stays. But nothing in the usual answers says why exactly the residual stream and attention for token mixing were never swapped out, while the MLP kept its role but changed its form.
- **Why did the adopted changes take the forms they did?** Pre-normalization, normalized queries and keys, capped logits, sinks, registers, output gates, log-N scaling with context length and hybrids at fixed attention fractions are small and specific. No usual answer addresses why these forms were adopted, and not others.
- **Where can the design still move, and what must a successor supply?** The usual answers guess from costs: inference, the memory rent of a long context, the data wall. They do not say which parts of the design can move, or what a successor must still supply.

The collection is silent on much that the usual answers cover: scaling laws and their economics, data, capital and hardware prices, in-context learning beyond what single heads and short compositions do, chain of thought, and what the MLP computes. The conditions account in Section 3 is complementary to the usual answers: hardware sets the prices of operations, and the conditions on the block are what is being priced.

Table 1: The usual answers and what the collection adds: for each explanation, whether the collection supports, explains, refines, corrects, is adjacent to or is silent on it, and where in this paper the bearing is argued.

The usual answer	Bearing	What the collection adds	See
Why this lineage won			
It fit the hardware: large dense matrix multiplications	refines	Hardware sets the prices; the collection says what is being priced.	Section 3.3
Scaling laws made capability predictable and purchasable	silent	The collection has no scaling-law results.	—
The ecosystem co-evolved around it	refines	Not disputed; each alternative that relaxes a constraint of the block must also re-supply the protection it gives up.	Section 3.3
Proving an alternative at scale costs too much	refines	Compensation (spare capacity elsewhere buying back a deficit) predicts that deficits cost less at scale (pending).	Section 3.4
History: the right objective, data and hardware in one place	refines	The path was historical; the endpoint looks constrained, as a reading.	Section 3.6
We measure what transformers are good at	adjacent	Diagnostics invariant under a natural involution (such as time reversal or swapping two classes) cannot see direction or contrast.	Section 4.1
In-context learning turned one model into a platform	silent	Outside the collection.	—
Why this design			
Weak assumptions plus abundant data win	corrects	The premise is wrong in a precise sense: attention's bias is averaging, with failure modes the rest of the block exists to prevent.	Sections 3.1 and 3.2
Training is parallel, and every position gives a signal	silent	Beyond this, the collection notes only that contracting recurrences also parallelize and that wall-time claims need verified kernels.	—
Exact, content-addressable memory makes recall possible	supports	Supported and sharpened: a linear memory recalls N arbitrary values exactly only if $N \leq d_k$, its key dimension; recall rests on one condition of six, the routing budget.	Sections 3.2, 3.7 and 4.4
The residual stream is a bus; blocks communicate, then compute	explains	The residual is the anchor, the one protection that capacity cannot buy back at bounded gain (bounded sensitivity) under strict mixing; the MLP is the writer.	Sections 3.2 and 3.4
It trains: conditioning is half the design	explains	"It trains" becomes two conditions: the anchor (Pre-LN, critical at initialization) and gain control.	Section 3.2
Data-dependent routing built from matrix multiplications	refines	Part of the price list; other hardware would price other machines differently.	Section 3.3
Softmax can sharpen into a pointer	refines	Refined: that a winning score's margin must grow like $\log N$ over N keys is known (Nakanishi, 2025; Veličković et al., 2025); the collection adds its abstention side and the joint budget; with independent scores whose likelihood ratio rises with the score, no router beats the argmax on the same scores; sharpness trades against abstention.	Sections 3.2 and 4.4
The architecture is not the algorithm	explains	This is compensation, measured at small scale, with an exception this answer does not anticipate: the anchor.	Section 3.4
Why it stays			
It absorbs its challengers as components	explains	Most of what gets absorbed into the mixing layer re-supplies a named protection (a reading); mixtures of experts, a change to the MLP, lie outside the lattice of mixing operators.	Section 3.6
Challengers converge towards attention	explains	Within the declared model classes, successors must supply the same conditions and retrace the lattice of mixing operators.	Section 3.7
Weaknesses get worked around above the architecture	adjacent	State tracking beyond counting needs non-commuting transitions (Grazzi et al., 2025); otherwise the collection is silent.	—
The design space around it is flat	explains	Explained and measured at small scale: compensation and tests of explicit knobs for protections the block already has.	Section 3.4
Scale sorts architectures, so far in the transformer's favour	refines	A mechanism, as a hypothesis: compensation grows with capacity (pending).	Section 3.4
The transformer endures as a template for learned computation	refines	What is fixed is the set of conditions and the cheapest current way to pay them (a reading).	Section 3.3

2.5 Attention's strong bias is averaging

One premise in the usual answers, that attention's inductive bias is weak, is wrong in a precise sense. Attention's bias about *which* tokens interact is weak; its bias about *how* they interact is strong. A softmax head averages. Its outputs are convex combinations of its values, so they stay inside the values' convex hull, the smallest convex region that contains every value (M§14.3). Applied again and again through depth with one fixed operator, averaging pulls every token toward a consensus unless a share of the signal keeps taking short paths, whose number of averaging steps stays bounded (M§§2.2, 2.3). Because each row of attention weights sums to one, a head cannot abstain (choose to contribute nothing) independently of its values (M§8.1). Read as a random walk that steps from token to token with the head's attention weights as probabilities, a head without circulation (net traffic around token cycles at equilibrium) can swap pairs of tokens but cannot shift tokens around a cycle. Along an open sequence, such a head can route forward reliably only by piling stationary mass (the equilibrium probability assigned to each token) exponentially toward one end (M§10.1). Much of the transformer block exists to counter the failure modes of averaging, which is where Section 3 begins.

3 A sweet spot of conditions

This section makes the collection's case for why the transformer persists, one step at a time. Each step states its claim, its strongest evidence with its evidence label, and its scope. The steps build on one another: the object, averaging rebuilt from its input and iterated (Section 3.1), fixes the conditions (Section 3.2), and the standard block is read, under a price model, as the sweet spot where all the conditions are cheap together (Section 3.3). The remaining steps help explain why the design has stayed there and show where it can still move.

3.1 The object: averaging, rebuilt from its input and iterated

A token-mixing layer computes, at every position, a weighted average of value vectors. Its weights form a row-stochastic kernel $P(X)$ built from the current representations X, so the layer is an averaging operator rebuilt from its own input, and the block applies it again and again through depth, inside a residual path. Two proved facts turn this description into a tool. First, the polynomial filters that remain averaging operators for every row-stochastic operator (combinations of its powers, as through depth, restarts or a residual path) are exactly path lotteries, random walks of random length, $f(P) = \mathbb{E}[P^K]$ (M§1.3). Thus what a design keeps through depth can be computed from its law of path lengths. Second, the outputs of nonnegative mixing with unit row sums stay in the convex hull of the values (M§§11.1, 14.3). This is why a head must deliver a mix of its values even when nothing is relevant, and why a frozen average cannot amplify: with its weights held fixed, no output moves further than the largest change among the values (M§1.4).

The same object appears in graph propagation, label propagation, adaptive denoising and test-time mixing, and the same laws govern it there. One path-lottery law covers oversmoothing in graph networks (Oono & Suzuki, 2020), the decay of label propagation and the rank collapse of pure attention repeated with one fixed operator (Dong et al., 2021), each known on its own (M§§2.3, 2.5, 4.1). On the propagation filters realized on a citation graph, the collapse-ratio identity (which gives the ratio of consensus gain to content gain from the path-length law) holds to machine precision. By contrast, a four-layer perceptron, a negative control because it does not mix across samples, shows no collapse across samples. This control is weak: it has one seed, and the deeper perceptrons tested did not converge, their final-layer features concentrating (M§4.1).

When the mixing weights are rebuilt from the state, a second law governs stability. Consider a loop that retains a fixed share of its initial input (the anchor) at every step and averages the rest with a stochastic operator rebuilt from the current state. One anchored-contraction certificate shows that this loop contracts to a unique fixed point when its sensitivity bound holds, meaning that the rebuilt weights change slowly enough with the state relative to the anchor share. Such loops include regularization-by-denoising with a nonlocal-means denoiser and an identity observation operator. The certificate's proof is checked in Lean. p-Laplacian graph networks, the graph-diffusion model nonlinear GRAND++ and deep equilibrium transformers share this feedback between state-rebuilt weights and a retained input, but need bounds on their additional terms

(M§§4.2, 7.1). The certificate extends to restart shares that depend on the state, the first of these additional terms: the cost in the contraction margin is how fast the share moves with the state times how far the mixed state can sit from its anchor (M§7.1). Beyond convergence, laws of the same operator govern propagation over frozen features, test-time context mixing and, in unreleased work, graph-consistency regularization (M§§4.2, 4.4, 12.4, 12.5).

These laws do not depend on the domain, which is consistent with the design travelling across domains, though it does not say why attention, rather than another averaging design, was the one that travelled. The claim is not that averaging explains every domain's success, nor in-context learning; heads also compose, and the MLP writes new features. The laws above are stated for one operator, or one adaptive map, applied repeatedly, while a trained stack applies a different operator at each layer; Section 5 says which statements carry over.

3.2 What fails without each of the six conditions

An averaging layer that is rebuilt from its input and iterated faces six failure modes. Each has a protection that is proved within a declared class and a signature that has been measured when it is removed (Table 2). These are conditions on a class of models, not six mechanisms that every successful model must implement: they say what goes wrong, and at what price, when a model in the class lacks the property (M§14.1).

The anchor: keep short paths. Averaged again and again, every token drifts toward a common average. A residual stack is a sum over routes: at each layer a token's signal is either averaged or passed on, so depth acts as a distribution over how many averaging steps a signal takes (a path law), and a pattern that one averaging step shrinks by a factor μ keeps $\mathbb{E}[\mu^K]$ of itself after a random number K of steps (M§1.3). For normalized, nonnegative path laws, a non-consensus mode (a pattern that distinguishes tokens) with eigenvalue $0 < \mu < 1$ keeps at least a fixed positive fraction of itself at every depth if and only if a fixed fraction of the paths stays within a bounded number of steps. In this class, when the per-layer gains on that mode stay bounded away from zero, the stack's gain on it is $\exp(-\Theta(S_L))$ in the mixing budget S_L, the expected number of layers that mix (M§§2.2–2.4). A residual connection supplies short-path mass. It keeps a fixed share only when the mixing budget stays bounded, for example when each branch shrinks with total depth. Pre-LN sits on the critical line between a floor and exponential decay at initialization, losing modes polynomially with depth (Section 3.3). Removing the identity path sends a minimal block to exact chance at depth 12 and beyond, though a constant output does not by itself separate collapse from a failure to train (M§5.2). Post-LN, whose mixing budget grows linearly with depth, has a trainability cliff from depth 18 on CIFAR-10, bimodal across seeds. A weight-tied block trained at four loops keeps 0.904–0.959 accuracy at 96 loops when it re-injects its input, against 0.023–0.032 when it does not (M§5.2).

Alignment: a free content path. Deciding which tokens to read and deciding what to carry from them are different jobs. An independent value map is sufficient to keep them apart. It is also necessary on an identified block model when tied values inherit the keys' nuisance, as exact tying does, because values that factor through the key map carry nothing the keys lack (M§6.1). Neither order of amplification and projection is safe on its own: amplifying before a projection exposes the part of the signal that escaped it to whatever the projection reads, and amplifying after a projection leaves in the output a component the projection would have removed. A separate value map closes the first route, for nuisance carried in the features, but not the second (M§6.2). In trained one-hop models on a synthetic routing-and-content task with value noise, a value map tied to the keys fails at head dimension d_b, the dimension of the content, with mean retrieval 0.29 against 0.97 for separate maps (M§6.1). Unlike the anchor, this deficit can be bought back (Section 3.4).

Gain control: hold the feedback below the anchor. Attention's weights are computed from its own outputs, so it can reinforce itself. An anchored adaptive iteration applies one weight builder (the map from current states to attention weights) repeatedly, as in a weight-tied loop, with anchor share β and mixing share $\alpha = 1 - \beta$. It contracts when the anchor outweighs the feedback, $m = \beta - \alpha\kappa_f > 0$, where κ_f is the feedback's loop gain. The threshold is sharp over the class, and the proof is checked in Lean. For tied dot-product softmax, the feedback bound is $\kappa_f \le 2sR^2$ (projected scores multiply it by the projections' norms), where s is the logit scale and R the radius of the representations; the bound needs that radius, since dot-product

attention is not globally Lipschitz (Kim et al., 2021) (M§7.1). Past the margin a smooth builder can have two attracting fixed points, and at the largest response of a synthetic search the equilibrium's input response (its sensitivity to input changes, in the Euclidean norm) rises from 1.77 with frozen weights to 18.9 when the weights adapt (M§§7.1, 12.2); a frozen average can exceed one in that norm, though no output moves further than the largest change among the values. Normalization bounds the radius but is not by itself a certificate. The per-layer bound on κ_f holds in any stack; contraction is a statement about the iterated map (Section 5).

Abstention: a way to deliver nothing. Softmax weights add up to one, so a head must deliver a weighted mix of its values even when nothing is relevant (zero only when they cancel): its output is a convex combination of them. A uniform logit shift, the change that would mean "attend less", leaves these weights unchanged: it lies in the kernel of the softmax Jacobian. Every zero-value patch that lets a head abstain (a sink token or a register whose value is zero, an off-by-one null or a learned sink logit) factors as a gate times an ordinary softmax, and so does an output gate. The gate controls how much the head delivers, while the softmax selects the mixture. Sigmoid attention lies outside that class (M§8.2). Standard causal models have no explicit way to abstain, but the causal mask makes the first token absorbing, and they route heavily to it (Xiao et al., 2024; Gu et al., 2025; Barbero et al., 2025): in seven public causal language models at 64 tokens, the minimum row mass sent to the first token has median 0.416 over 2,700 captured operators, heads and layer averages, each averaged over 16 inputs, and a per-input census of GPT-2 small gives a per-head median of 0.080 (M§2.7). Letting rows abstain changes the sinks. Sigmoid attention in a small vision transformer, with mean row mass near one third, cuts the share of layers with a sink from 0.53 to 0.22 at depth 12 (lower in two of three seeds) and from 0.65 to 0.07 at depth 24 (three seeds each), at a cost of 2.5 and 1.7 points of test accuracy, paid in every seed, and an output gate more than halves the mean attention paid to low-value content tokens in a small causal language model, lowering it in each of three seeds (M§8.5). The output gate's readout carries a caveat: because the first token absorbs a causal walk, the stationary-mass test of the sink detector is vacuous on causal operators, so this readout rests on value norms.

The routing budget: a margin that grows like $\log N$. The more tokens compete, the larger the score advantage the right one needs. A single key competing with N others at a common logit needs a margin of $\log(N(1-\eta)/\eta)$ to keep a share $1-\eta$, so with a bounded logit scale retrieval fades as the context grows, as is known for retrieval alone (Nakanishi, 2025; Veličković et al., 2025). With a null that absent and present queries share (queries without and with a matching key), the largest number of background keys that keeps abstention a^* and retrieval w^* at once is the exact budget $N^* = e^{2\tau}(1-a^*)(1-w^*)/w^*$ for content logits bounded by τ (M§9.1). On a synthetic retrieval task trained at three lengths, the context length at which accuracy drops (the accuracy onset) grows as $e^{2.14g}$ in the trained logit gap g, a descriptive fit over fifteen models whose onsets are fixed by their three training lengths. That is faster than the weight bound's e^g, because a learned decoder can read a match whose weight has already fallen (M§9.2). Along the image-resolution axis, where upsampled images with interpolated positions increase the patch count, a training-free $\log N$ re-scoring recovers 0.65 and 0.83 points in vision transformers trained from scratch at one resolution, 3–5% of the accuracy lost; a fitted constant temperature matched that unfitted rule and beat a $\log N$ rule with a fitted exponent (M§9.6). In ViT-S models trained on ImageNet-100, tempering every score in the same way raised accuracy at sixteen times the training token count by at least 3.7 points in every seed of the sink rules, as a prespecified test confirmed, and tempering plain softmax gained as much (M§17.4). No result says that capacity elsewhere supplies this margin. Inside their training length, trained causal models of 0.6 million and 125 million parameters meet it through position. With rotary positions the log of the background sum (the denominator sum over non-matching keys) grows far more slowly than $\log N$, because the positional distance kernel holds distant scores down. Beyond that length this lasts while the positional treatment keeps positions in the trained range (M§9.2); otherwise the logit scale or a $\log N$ scaling must supply the margin. When a head is no wider than the content it must recover through its keys, the budget is also dimensional (M§6.1).

The arrow: a circulating component in each head. Many tasks need information to move one way along a sequence, as when a head reads the token that follows a match. Viewed as a Markov chain on the tokens, a head that moves information forward reliably along a sequence needs a circulating component (net probability flow around cycles), or must pile its stationary mass exponentially toward one end. For a head

without circulation (a reversible chain with a positive stationary law), the odds of moving to a successor rather than a predecessor have geometric mean exactly one around every cycle of tokens whose successor and predecessor probabilities are positive: it can swap tokens but not shift them around a cycle (M§10.1). This is Kolmogorov's criterion for a reversible chain (Kolmogoroff, 1936; Kelly, 1979), read as a statement about what a head can do. Tying queries to keys costs more than circulation. Tied scores form a positive-semidefinite Gram matrix, which on geometric average around a cycle cannot prefer a neighbour to the reading token itself. On a synthetic cyclic-retrieval task with small models, tied models reach 60% and 76% successor retrieval at depths 2 and 4, against 98% for circulation-free heads with a symmetric indefinite score form, which is not subject to that Gram restriction, and 100% for standard ones. Under a causal mask, which gives look-back relations their direction, the Gram form still copies less surely than untied scores at varying offsets in a small synthetic test, with cross-entropy 3.4 times higher on average over five seeds and accuracy 0.8 points lower, while a bidirectional lookup stays at parity (M§10.1). Reversing the first layer's learned circulation in trained retrieval models turns successor retrieval into predecessor retrieval (M§10.5). The statement is per head: compositions of reversible heads can acquire direction, and decoders take theirs from the causal mask.

The writer lies outside the inventory. Moving information around is not enough. Averaging cannot create new features: a mixing step with row-stochastic weights, however they are chosen, keeps every output in the convex hull of its input rows and cannot raise their rank, while a shared pointwise nonlinearity can do both (M§11.1). Replacing averaging with orthogonal token transport hits a similar limit through an invariant: mixing the tokens of a group by one shared orthogonal matrix preserves the group's channel-by-channel inner products (its channel Gram matrix), so a readout that depends only on that matrix cannot change, though individual tokens can. A transport-only mixer of this kind reaches a seed-mean dictionary-retrieval R^2 of 0.18–0.26 at the training size of 16 associations (below 0.03 at 64), against at least 0.99996 for standard and additive reversible blocks at 8–64 associations, and loses 10 points on CIFAR-100, with arms that are not parameter-matched; adding a nonlinear feature-writing channel to the transport mixer restores classification but not retrieval (M§§11.3, 11.4). In a transformer the MLP and W_O write, and the collection has no theory of what the MLP should compute (M§11.7). Stated for routing machines in general, writing becomes a seventh condition (Bruce, 2026e) (M§19.1).

Scope. Each row of Table 2 is a theorem within its class plus a measured removal signature, and this is the strongest step of the case. Its limits are specific. The classes are particular (the anchor, for example, is stated for normalized nonnegative path laws of a fixed mode), the removal signatures were measured at small scale, mostly below 25 million parameters, and the gain-control signature is synthetic. A row would be falsified by a trained model in its class that lacks the protection and shows no signature; for the anchor, a registered test of exactly this is pending. Compensation (Section 3.4) does not excuse a missing signature by default. It is itself a measured claim with a named currency, such as MLP width, head width or head count: a model that lacks a protection and shows no signature counts against its row unless removing that currency brings the signature back. That test exists so far for the value map; for the anchor, compensation at bounded gain is ruled out under strict mixing, and the trained test is the one pending above.

3.3 The sweet spot

Each part of the standard block pays for one condition (Figure 2). The residual stream, with normalization in front of each sublayer, is the anchor. It sits on the critical line at initialization, where a logarithmic mixing budget gives polynomial rather than exponential decay: in the mean-field energy model, a pre-normalized stack retains $L^{-s(1-\mu^2)}$ of a mode through L layers, where μ is the mode's eigenvalue and s, here, the attention share rather than the logit scale. At width 512 (three seeds), random pre-normalized transformers with bidirectional attention, whose attention is nearly uniform at initialization, match the predicted exponent $-s$ with a largest deviation of 0.041, and, at width 512, linear token-mixing stacks match its dependence on μ within 0.05. Post-LN's mixing budget, by contrast, grows linearly with depth and drives exponential decay (M§2.6). For alignment, the value map W_V is the free content path. LayerNorm, and later QK-norm and logit caps, control gain, by norm control rather than a certificate. Sinks, registers and output gates patch abstention inside unit row sums. A fixed logit scale supplies the routing budget within the trained length, helped in trained causal models by rotary positions that hold down background growth. Beyond the

Table 2: Six conditions and the writer: what each guarantees, what fails without it (record and signature), how the pre-normalized softmax block pays for it, whether capacity elsewhere can compensate, and the chapters of the monograph that develop it (Bruce, 2026a).

Condition	What it guarantees	What fails without it	In the block	Compensated or patched?	Chapter
Anchor	A mode keeps at least a fixed fraction of itself at every depth iff a fixed fraction of paths stays short; with per-layer gains on that mode bounded away from zero, the stack's gain on it is $\exp(-\Theta(S_L))$ in the mixing budget	Identity path removed: exact chance at depth ≥ 12, which does not isolate collapse. Post-LN cliff at depth ≥ 18. Looped block trained at 4 loops, at 96 loops: 0.023–0.032 without re-injection, 0.904–0.959 with it	Residual stream with Pre-LN, critical at initialization	No, at bounded gain under strict mixing	2, 5
Alignment	An independent value map is sufficient, and necessary on an identified block model when tied values inherit the keys' nuisance	Tied value map (with value noise): retrieval 0.29 vs 0.97 at head dimension d_b	The value map W_V	Yes: width (0.96 at $2d_b$) and MLP capacity	6
Gain control	Contraction when $m = \beta - \alpha\kappa_f > 0$, with $\kappa_f \leq 2sR^2$ for tied dot-product softmax; sharp over the class	Past the margin, two attracting fixed points (synthetic); at a synthetic search's largest response, Euclidean response 1.77 frozen vs 18.9 adaptive	LayerNorm, later QK-norm and logit caps: norm control, not a certificate	Partly; norm control is not a certificate	7, 12
Abstention	Rows summing to one cannot abstain independently of the values; every zero-value patch for abstention and every output gate factors as a gate times a softmax	Sigmoid attention (row mass 0.34 and 0.31 at depths 12 and 24) cuts sink incidence from 0.53 to 0.22 and from 0.65 to 0.07; an output gate halves content-sink mass (value-norm readout; causal caveat)	Sinks, registers and gates, patched inside unit row sums	Patched; at long context no fixed rule with a null shared by absent and present queries keeps both abstention and retrieval	2, 8
Routing budget	One key among N needs a score margin growing like $\log N$; with a shared null and logits in $[-\tau, \tau]$, $N^* = e^{2\tau}(1-a^*)(1-w^*)/w^*$	Accuracy onset grows as $e^{2.14g}$ in the logit gap g; a training-free $\log N$ re-score recovers 0.65 and 0.83 pp (3–5% of the loss, matched by a fitted constant) on upsampled images	A fixed logit scale, helped by position within the trained length; beyond it, positional interpolation and log-N scaling, patched	Within the trained length, through position; no result for capacity elsewhere; dimensional at narrow heads	9, 17
Arrow	A head without circulation can swap tokens but not shift them (per head)	Tied (Gram) scores: successor retrieval 60% and 76% at depth 2 and 4, against 98% for circulation-free indefinite scores and 100% for standard heads; under a causal mask, cross-entropy 3.4 times higher at varying offsets	Untied queries and keys; the causal mask in decoders	Yes, by composition across heads and layers	10
Writer (outside)	Averaging cannot create features: outputs stay in the inputs' hull, and rank cannot rise	Transport-only mixer: retrieval R^2 0.18–0.26 at 16 associations (below 0.03 at 64) vs ≥ 0.99996 at 8–64; CIFAR-100 −10 pp (parameters not matched)	The MLP	A write channel restores classification, not retrieval	11

trained length, positional interpolation patches the budget. Log-N scaling, added later, helps on interpolated positions unless the interpolation rule already applies a log-length temperature (Section 6.1). Untied queries and keys, and in decoders the causal mask, supply the arrow. The MLP writes. The inventory was assembled by asking what each part of the block protects, so this match holds by construction; what does not follow from the construction is each row's necessity within its class (Section 3.2) and the predictions about other designs (Section 6.2).

In small trained models, tuned standard components already carry the protections credited to them and capabilities that dedicated knobs are added to supply. In graph transformers, a pre-normalized stack's last-layer anchor coefficient, an input–output regression coefficient, is 0.998 on average, against 0.934 for Post-LN (165 cells each: ten datasets, depths 2 to 16 with incomplete coverage, five split seeds), and a tuned Switch load-balancing loss already balances its experts (three seeds) (M§§5.3, 15.4). Multi-head attention with signed output maps already subtracts across heads: in a single small block trained for 500 steps, difference retrieval (recovering the difference of two stored values) reaches a mean R^2 of 0.46 with one head and 0.998 with four (25 matched runs) (M§15.5).

We read this as a sweet spot: under the prices of parallel hardware and gradient descent, the pre-normalized softmax block is the one known point where all six conditions are cheap together. This is a judgment, not a computation. The price model's formal parts price spectral costs (how much collapse toward consensus each species of operator must pay to suppress an unwanted mode (M§14.4)) and budget allocations across protections under assumed concave prices (M§§16.1, 16.2). Cheapness on hardware is judged, and the alternatives' prices were not measured on a common budget of hardware and training (M§14.5). Nor are the prices independent of the design: hardware, kernels and recipes co-evolved with the transformer, so part of its cheapness is incumbency. The transformer is the cheapest point under one price list, not the only design that meets the conditions: a catalog of 43 families of routing machines shows how others pay for them (M§§14.4, 19.2). This is also where the account meets the usual answers of Section 2: hardware sets the prices; the conditions are what is being priced.

Underneath the price list sits a family of lower bounds (M§9.5). Whatever the machine, naming one of N stored items costs about $\ln N$ nats, by Fano's inequality (Cover & Thomas, 2006), and keeping a context of unstructured items for arbitrary questions asked later costs memory in proportion to the number of items. Machines pay in different currencies: key width (dimension), sharpness of the read (logit range), state or order. Each bound holds within its own class of machines and prices one currency there; the bounds set no exchange rate between currencies. Together they say that no fixed allotment of dimension, logit range or state survives a growing context, which may be one reason efficient alternatives keep reacquiring attention-like memory.

Relaxing any constraint of the block buys a capability and obliges the design to re-supply the protection it gave up (M§14.2). Sigmoid attention gains freedom in its row sums and must re-supply a length budget and norm control; recurrences gain a bounded state and must re-supply recall; signed mixing gains subtraction and must re-supply a cap on its norm or per-head normalization (M§§14.2, 14.3).

3.4 Why the neighbourhood is flat

Two measured mechanisms make local changes rarely pay: when a change weakens a protection, spare capacity elsewhere can buy back the deficit (with the anchor as the exception), and when a change turns a protection into an explicit, adjustable setting (a knob), a tuned block already supplies it.

Compensation. Each protection's necessity is proved on a construction built to isolate it, and a trained network evades the theorem whenever a resource, such as width, heads or MLP capacity, is spare (M§15.1). On a theorem's own construction for $C = 10$ classes, a content map of rank $C - 2$ sits at 90.16% against a predicted ceiling of 90.0%, while a content map of the same rank, trained jointly with its readout, reaches 99.84%; both are means over five seeds. Where a single nonnegative head cannot subtract, multi-head attention buys back subtraction across heads (Section 3.3) (M§15.5). A value map tied to the keys fails at head dimension d_b (the content dimension of Section 3.2), and twice the width restores retrieval to 0.96. At depths 12 and 24, MLP capacity also absorbs the accuracy drop (tax) of tying the value map. In small

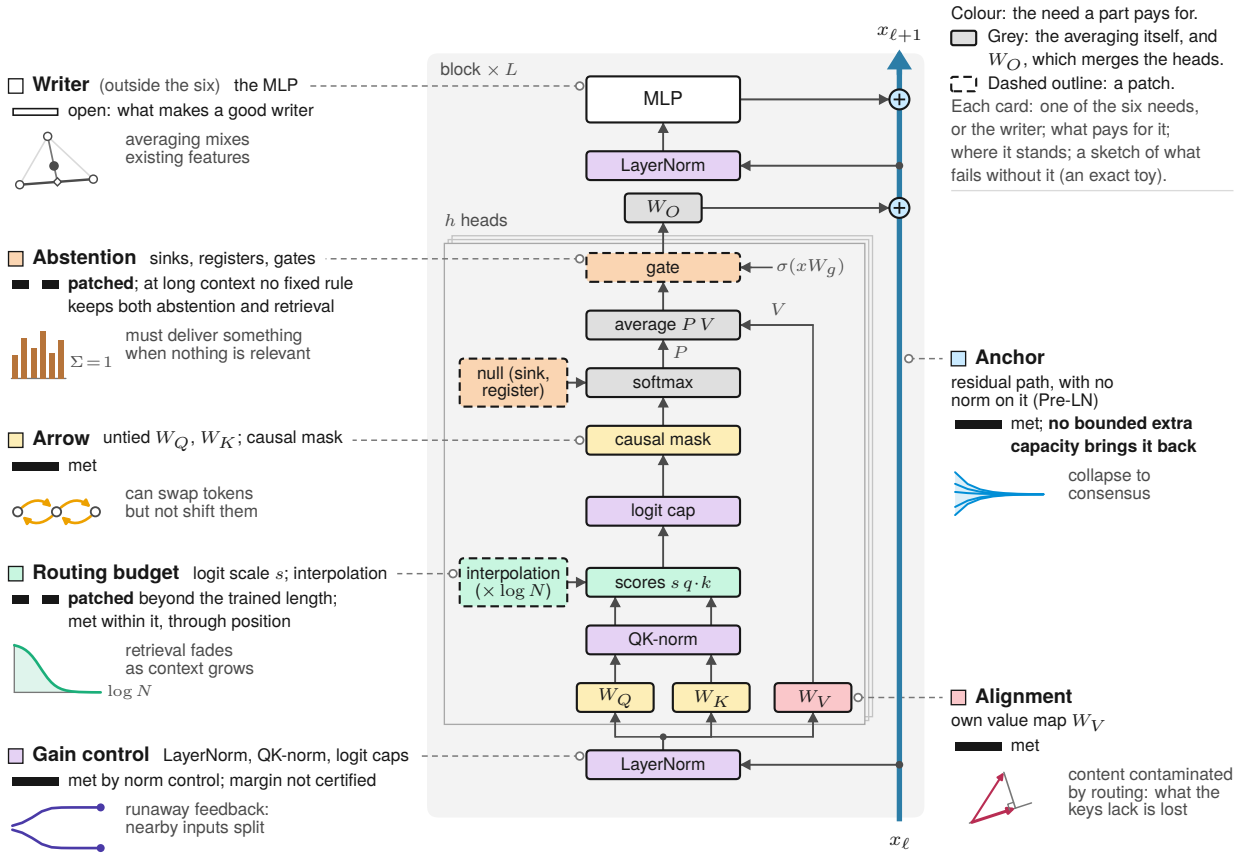


Figure 2: **Each of the six needs and the writer is paid for by a familiar part of the pre-normalized block; two of the needs, abstention and the routing budget, are patched rather than met at long context.** As in Figure 1, inside is what pays for each need and outside is what fails without it. Block ℓ of L maps the residual stream x_ℓ to $x_{\ell+1}$; s is the fixed logit scale, N the context length, and x in the gate $\sigma(xW_g)$ the normalized input; positional encoding is not drawn. The patches are positional interpolation beyond the trained length (Peng et al., 2024), with log-N scaling, drawn in parentheses, where positions are interpolated (Chiang & Cholak, 2022); a value-null option at the softmax (a key that takes weight and delivers nothing), such as a sink or a register (Xiao et al., 2024; Darcet et al., 2024); and an output gate on each head (Qiu et al., 2025). With bounded logits and a null that absent and present queries share, no fixed rule keeps both abstention and retrieval at long context (M§9.1). Each sketch is an illustrative construction: an exact toy of the failure beside it, not a measurement. Each need, and the writer, is proved within a declared class and has a signature measured when it is removed, synthetic for gain control (Table 2). Under strict mixing, no extra capacity of bounded gain brings back a missing anchor (M§5.5). Gain control is met by norm control, not a certificate (M§7.5). The arrow is stated per head. What makes a good writer is open. One part per need is a compression: the logit scale also enters the feedback gain and W_O also writes (M§§7.2, 11.7), and QK-norm (Henry et al., 2020), logit caps (Gemma Team, 2024) and the patches are later additions; models differ in which they have. That the block pays for all the needs, cheaply and at once, is an interpretation under a price model.

vision transformers at depth 24 and width 192 on CIFAR-100, that tax is +1.70, +0.04 and −0.72 percentage points at MLP ratios 0, 1 and 4. The penalty shrinks as the MLP widens: the interaction between tying and capacity is negative in all four configurations at depths 12 and 24 (Holm-adjusted $p \leq 8.4 \times 10^{-6}$), while at depth 4 the tax stays positive over the measured range; the break-even MLP ratio where the tax reaches zero is 1.11, with a stratified 95% bootstrap interval of [0.93, 1.63] (M§6.6). These results come from a reanalysis of a known outcome, over five initialization-only seeds that share one data order. The comparison is not parameter-matched: tied models have fewer parameters. This is compensation at the level of accuracy; the records do not show the mechanism. In theory, the condition for a downstream compensator to exist is settled: inputs sent to the same state by the weakened (restricted) map must have the same target value, meaning that the target is constant on the fibres of the restricted map (M§15.1). In a trained model, an absent removal signature is put down to compensation only when removing the spare resource brings the signature back, the test stated in the scope of Section 3.2.

The anchor is the exception. It is the protection whose loss is proved beyond the reach of capacity elsewhere: under strict mixing, capacity of bounded gain cannot buy it back. If the layers act on a common discriminative invariant subspace with norms $q_\ell \leq 1$, any downstream recovery map with Lipschitz constant G that must separate two distinct inputs in that subspace by $c > 0$ times their distance needs $G \geq c/\prod_\ell q_\ell$ when the product is positive. Under strict mixing, where these products vanish with depth, no compensator of uniformly bounded gain recovers the lost distinctions; large gain at finite depth, an operator close to the identity or an external input can recover them (M§5.5). Trainability already explains why the residual stream is the part of the block no one has replaced; this non-compensability adds a second reason.

An explicit knob does not beat the block's own mechanism. Twelve interventions, each registered in the repository before its runs (Appendix B), turned one protection into an explicit, certifiable knob inside a standard model: certified fast weights, an anchored mixture-of-experts router, sign-flexible filters, a sink-anchored decoder and eight more (M§15.4). Each knob's mechanism held in several rows where its hypotheses held, several of them to numerical precision, but not in all: it was not met in one row, was inconclusive in two, and its falsifier was realized in a fourth. Certified fast weights, for example, had no divergences in 48 certified runs, where 53 of 273 uncertified runs diverged (41 of 45 aggressive and 12 of 228 tuned, the tuned ones all in language modelling, which had no certified arm) (M§18.2). Yet each of the ten capability tests that could be scored was null or negative; the row-by-row record is still to be released. Where a knob gained, the gain stayed below the preregistered bar or was paid for on another axis. The sink-anchored decoder, a 76.7M-parameter model trained at 512 tokens, bought robustness at length, cutting perplexity at 2,048 tokens from 88.1 to 59.8. This came at an in-distribution cost: perplexity at its training length rose from 25.80 to 26.85. These are means of two seeds, which agree in direction. The study's explanation, measured directly so far only for the anchor and a Switch router's balance (Section 3.3), is that a tuned baseline already supplies the protection.

Measured at small scale, these two mechanisms together offer an answer to the first open question of Section 2.4: a reason most published modifications barely move accuracy. If compensation grows with capacity, it would also explain why the small-scale gains of alternatives fade with scale. That is a hypothesis, and its test is pending. *Pending:* Compensation budget at scale (vision transformers on ImageNet-100 and a 125M language model): does the price of a deficit shrink with capacity? *Pending:* The anchor non-compensability sweep in trained networks.

3.5 Do designs jump rather than drift?

This step is a hypothesis about why designs would jump between discrete choices rather than drift smoothly. Under concave prices, some best allocation of a design budget across protections is a corner of the feasible set (a vertex), with at most one protection strength strictly between its bounds. For a fixed feasible set under those prices, the best corner changes only when the costs of competing corners cross, so the optimum jumps from one corner to another (M§16.2). That a concave cost is minimized at a vertex is classical (Rockafellar, 1970, §32). In 36 constructed instances an interior split never beat the best corner, though knowing the optimum is a corner does not say which corner it is: under capped, anchored pricing, greedy filling chose a suboptimal corner in 5 of 24 instances.

In deployed architectures, hybrids that mix full and linear attention do sit at or next to small-denominator fractions of attention, with full attention in 12 of 48 layers of Qwen3-Next and 7 of 27 of Kimi Linear (Qwen Team, 2025; Kimi Team, 2025). But that record is descriptive, and it was compiled by those holding the hypothesis.

The vertex result's premise, concave pricing at the level of architectures, is an assumption, not a measurement. And on a single attention-fraction axis concave pricing predicts 0, 1 or a binding cap, never a fraction such as 1/4 in itself. The small-denominator fractions used in the predictions (Section 6.2) are a scoring convention, not a consequence of concave pricing. The deployed fractions therefore do not yet discriminate between corners and drift; a test needs measured prices, or a controlled comparison of jump against drift, which is planned (M§16.4).

3.6 Adopted changes in retrospect

Read retrospectively, many of the changes adopted in the mixing layer and its block over the last decade re-supply a named protection, and most of these read as responses to the failure each one fixes (M§16.5). Residual paths and pre-normalization re-supply the anchor and choose a mixing budget (He et al., 2016; Wang et al., 2019; Xiong et al., 2020). QK-norm, σReparam and logit soft-capping control attention's gain (Henry et al., 2020; Zhai et al., 2023; Gemma Team, 2024), and z-loss controls the output logits (Chowdhery et al., 2023). Sinks, registers, off-by-one normalization, learned sink logits and output gates supply abstention (Xiao et al., 2024; Darcet et al., 2024; Miller, 2023; OpenAI, 2025; Qiu et al., 2025). YaRN's interpolation and temperature, log-N scaling and scalable softmax pay the routing budget beyond the trained length (Chiang & Cholak, 2022; Peng et al., 2024; Nakanishi, 2025). In trained 125M language models, however, scaling every logit by $\log N / \log N_{\text{train}}$ (the ratio of log context length to log training length) helped only when positions were interpolated without another log-length temperature (Section 6.1). Differential attention is signed mixing with norm control, not a certified cap on the norm (Ye et al., 2025). Contracting recurrences and hybrids at fixed attention fractions bound the state and keep a reserve for recall (Yang et al., 2025; Lieber et al., 2024; Qwen Team, 2025; Kimi Team, 2025; MiniMax, 2025). Latent key-value caches compress the content channel (DeepSeek-AI, 2024); learned sparse attention relaxes density with learned support (Yuan et al., 2025; Lu et al., 2025); and looped models with input re-injection restore the anchor along the axis of recursion (Bansal et al., 2022; Geiping et al., 2025). The chronology behind Figure 3, from the paper released with this one on the transformer as a constrained equilibrium of protections (Bruce, 2026d), selects thirty-three sources from thirteen such groups of proposed or adopted changes and dates them by first public release; several, among them differential attention and looped models, are proposals that practice has not generally adopted (M§16.5).

This is a reading, not evidence. None of these changes was predicted from the conditions (Section 3.2) before it was adopted, so none counts in their favour. The lattice of trade-offs (Section 3.3) describes relaxed constraints and the protections that must then be re-supplied (M§14.2). It is silent on much that was adopted, such as changes to the MLP, to kernels, to the sharing of keys and values across heads (multi-query and grouped-query attention; Shazeer, 2019; Ainslie et al., 2023) and to the block's topology (parallel attention and MLP branches). We searched for a counterexample: an adopted change to the mixing layer that removes a protection without re-supplying it. We found none, but the search was ours, not blind. We include the reading because it answers the third open question of Section 2.4 (why the adopted changes took the forms they did), which nothing else does, and because it is the most intuitive way into the account.

The same reading runs forward as a blueprint. Assembling the smallest block that meets the inventory of Section 3.2 gives the pre-normalized transformer, within four corrections: explicit registers, one symmetric structure map where direction is not needed, a log-N budget, and residual branches scaled by $1/L$ (M§16.6). Since the inventory was read from the block, this shows that the account is consistent, not that it is independently confirmed. Because the symmetric structure map makes each layer reversible, in small models this minimal certified block keeps detailed balance to 10^{-12} on every layer of 50 trained encoders. The same block, completed with interleaved MLPs and pre-normalization, beat a parameter-matched ViT-Tiny after 100 epochs in 20 of 20 paired runs on CIFAR-10 and CIFAR-100, by 1.8 to 3.9 points averaged over the five seeds of each dataset and learning rate (M§16.6). At 22 million parameters, the recorded margins over ViT-Small were 10.59 and 8.02 points of top-1 accuracy on Tiny-ImageNet and ImageNet-100, averaged over

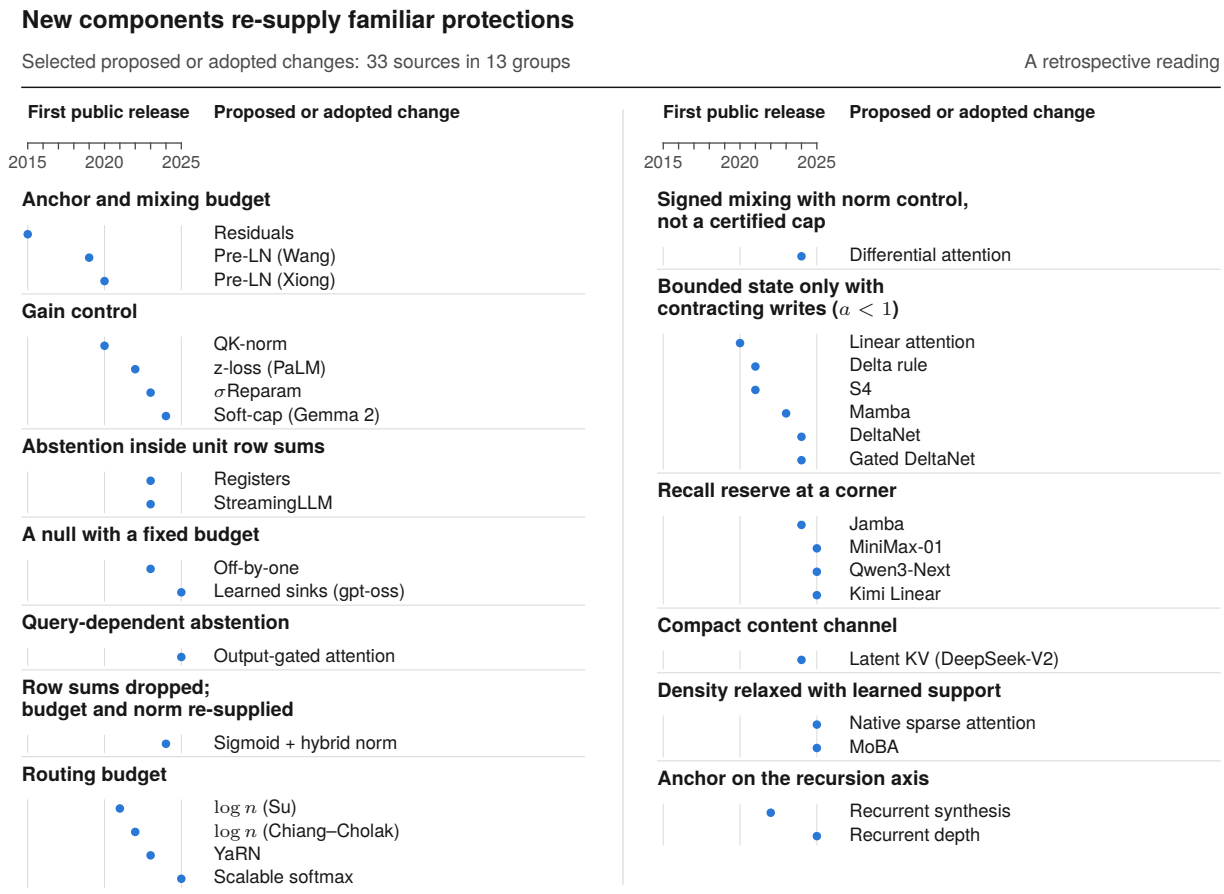


Figure 3: **Selected architectural changes revisit familiar protections: a retrospective reading, not evidence for the conditions.** The figure draws on a chronology of 33 separately dated sources in 13 groups (M§16.5). Each dot marks the year of first public release (arXiv v1, blog or model card), not adoption; each row names one source, and vertical position only separates entries. The two timelines use the same scale. Group headings give the readings of the paper on the transformer as a constrained equilibrium (Bruce, 2026d); the examples are proposed or adopted, not an exhaustive corpus or a claim of global priority. Differential attention supplies norm control, not a certified cap. Bounded state requires contracting writes ($a < 1$, for a decay factor a on the state); listing a method does not certify that condition. PaLM's z-loss acts on output logits, so its gain-control placement is a block-level interpretation. Besides the works cited in Section 3.6, the figure plots Su (2021), Ramapuram et al. (2025), Katharopoulos et al. (2020), Schlag et al. (2021), Gu et al. (2022), Gu & Dao (2024) and Yang et al. (2024). Exact source dates and their qualifications are kept with the figure's records.

five paired seeds. These margins are not evidence of dominance: the baselines were not tuned, and the tuned replication has not run (M§16.7). The path to the transformer was historical; its endpoint, read this way, looks constrained. *Pending:* The tuned replication of the minimal block at 22M parameters.

3.7 Where the design can still move

The room that remains is where a condition is patched rather than met, or paid at a price that another design could avoid (Figure 4): abstention at long context, bounded memory, decisions that read maxima instead of sums, and the writer.

Abstention at long context. Unit row sums forbid abstaining independently of the values, and every fix in use (a sink, a register or an output gate) sits in one cell of the design lattice. A two-sided law governs that cell. With content logits bounded by τ, no fixed, learned or length-calibrated null that absent and present queries share keeps both absent-query abstention weight a^* and present-query single-key retrieval weight w^* beyond $N^* = e^{2\tau}(1-a^*)(1-w^*)/w^*$. The logit range must grow. Tempering, which multiplies every score, the null included, by $\log N / \log N_{\text{train}}$, is one way that keeps both when the margins are strictly separated (M§§9.1, 9.2). The same budget decides any other schedule that scales the content scores and sets a null (M§9.1). In a controlled pilot of trained lookup-or-abstain heads, at logit scale at most 10 and a training length of 64 keys, a fixed learned sink's abstention falls from 0.874 to 0.007 (seed means) between 64 and 65,536 keys, along the law's prediction within 0.002 per seed; length-calibrated rules lose retrieval from 2,048 keys; and the tempered rule keeps both over the tested range, empirically rather than by certificate. In a registered extension to seven other settings, every rule that sums over the keys, tempering included, lost a side with heavy-tailed keys, and at the smaller budget tempering also lost abstention with clustered keys and with four heads (M§17.2).

Trained language models can offset a fixed sink's decay inside their training length by holding the background sum down through position. The prespecified production-model test, on gpt-oss with banded layers (whose finite window caps the background) as a built-in control, was undecided by its rule, although descriptively its slopes lean toward such compensation through position. Its books control, which compares next-token loss on natural and shuffled text, failed, and its sink log-odds slopes matched neither prespecified branch: confirmation of the law or compensation (M§17.3). In trained language models at 125M parameters, tempering every logit by $\log N / \log N_{\text{train}}$ on top of YaRN did not keep abstention and raised long-context perplexity. YaRN already carries a log-length temperature, and none of the probe's retrieval heads had the separated margins the schedule needs. Inside their training length, these models hold abstention nearly level through position: the rotary kernel holds the background's growth down, and training widens the kernel when the null itself grows with length, as small models with and without positions confirm. Beyond the training length, the hold on text lasts while positions are interpolated back into the trained range, and fails natively (without interpolation). On the probe's absent queries it did not hold even with YaRN: the fixed sinks kept about a tenth of their null mass at eight times the training length. Every rule that gives attention a null removed the first-token sink and most massive activations, as earlier reports found for heads that can do nothing, attention biases and output gates (Bondarenko et al., 2023; Sun et al., 2024; Qiu et al., 2025). At this scale no retrieval head reached the registered identification threshold (the largest weight was 0.247, against 0.3), so these abstention results describe attention weights, not behaviour (M§17.4).

Bounded memory. A fixed-size state stays bounded at every length when its writes contract by a common factor. A delta-rule write with decay contracts in this way, with bounded keys and values and a small enough step. Non-expansiveness alone does not ensure boundedness, and such a state can grow like $\sqrt{t}$ (M§18.2). In a trained hybrid a decay gate acts as the anchor that supplies this contraction: without it, the log of the state's norm rises by 2.23–2.28 over a sequence of twice the training length, in the layer where it rises most, against 0.24–0.87 across the four variants that keep it (five seeds each, at the larger of two model sizes) (M§18.1). A bounded state's recall price is a capacity, not a distance: a linear memory can recall every assignment of values to N keys exactly if and only if the keys have rank N, so exact recall of N arbitrary associations needs $N \le d_k$ per head, where d_k is the key dimension (M§18.3). A gated delta rule recalls 32 pairs at 0.998–0.999 (five seeds); at 128 pairs, beyond the 32 it was trained on, it recalls 0.78–0.83. These records hold no attention arm for comparison, so the recall cost of bounded state is not yet established in

trained models (M§18.3). Because bounded state caps exact recall per head, a hybrid keeps an attention reserve for recall, though even one layer with full context keeps the cache linear in the context (M§18.4). *Pending:* The qualified hybrid ratio sweep at 125M and 350M.

Maxima instead of sums. With an exchangeable background, a decision that adds up all N competitors at a fixed logit scale must be retuned as their number grows, by a $\log N$ schedule; a decision that reads only the largest needs none, and fails only when the background maximum reaches its threshold (M§§9.4, 17.5). In constructed tests, the fixed softmax sink failed at the predicted onset in four of four configurations. By contrast, a key check abstains unless its best score clears a threshold; with a weak match in these tests, it outlasted $\log N$ tempering. A head built on maxima, the max-statistic head, which routes by a softmax over its top keys and abstains through a gate that compares its largest score with a learned threshold (Bruce, 2026e), should then keep abstention and retrieval at every length without a schedule, against backgrounds bounded below its threshold. In the trained 125M language models, with positions interpolated by YaRN, its null and match weights both fell by less than half a nat in log-odds from the training length to eight times it, on the seed means though not in every seed, which no other rule achieved; five rules kept the match weight alone better. Without interpolation it failed as the matching key fell far down the ranking and the background maximum grew with length, hitting the evidence wall that the theory names, where background keys outscore the match (M§17.4).

The writer. No theory here says what a good writer is. The main open problem the framework leaves is how much of each protection's price the MLP, which writes, can absorb, and at what width (M§§11.7, 19.6).

What a successor must supply. Within the stated operator classes, any replacement must keep short-path mass, a free content path, gain control of the full feedback, a way to abstain whose budget survives length, a routing budget that grows like $\log N$, an arrow (from circulation, a mask or composition), and a writer (M§19.6). We expect a successor to win on a resource axis at matched quality, not on quality at matched cost.

3.8 How strong this case is

The evidence for the case in Section 3 ranges from theorems and measurements to interpretations and pending tests, in order of weight:

1. **Proved within declared classes.** The conditions and the laws are theorems about stated classes of operators or models, except the stationary law of the tempered head in Section 3.7, which is checked numerically, not proved. Most are also checked numerically, and several are formalized in Lean (Appendix B). None says that every model must implement six named mechanisms.
2. **Measurements are mostly at small scale.** Removal signatures and compensation are measured in controlled training runs, mostly below 25 million parameters and none above 125 million. For gain control, the removal signature is synthetic. Evidence at production scale comes instead from measurements on public checkpoints (Sections 4 and 5): 2,700 captured operators, sixteen pretrained language models, 184,320 certified head kernels and seven production vision transformers.
3. **Idealized settings.** Many results are proved for one operator applied repeatedly, or for idealized backgrounds and constructed keys; Section 5 labels each.
4. **An interpretation under a price model.** "All cheap at once" (Section 3.3) rests on a semi-formal price model, not on measured prices of the alternatives, and corner structure (Section 3.5) is a hypothesis under assumed concave prices.
5. **Retrodiction is not evidence.** The reading of the adopted changes in Section 3.6 is an interpretation.
6. **The test is what was stated in advance.** The account is tested by the predictions in Section 6.2 and the decisive tests of Appendix B.4. Two of those tests, both on long-context abstention (Section 3.7), have run. In the first, on a production model, the outcome was undecided by its prespecified rule: one of its two corpora failed its control, and its slopes matched neither prespecified branch, though descriptively they lean toward compensation through position. In the second, in trained 125M language models, tempering

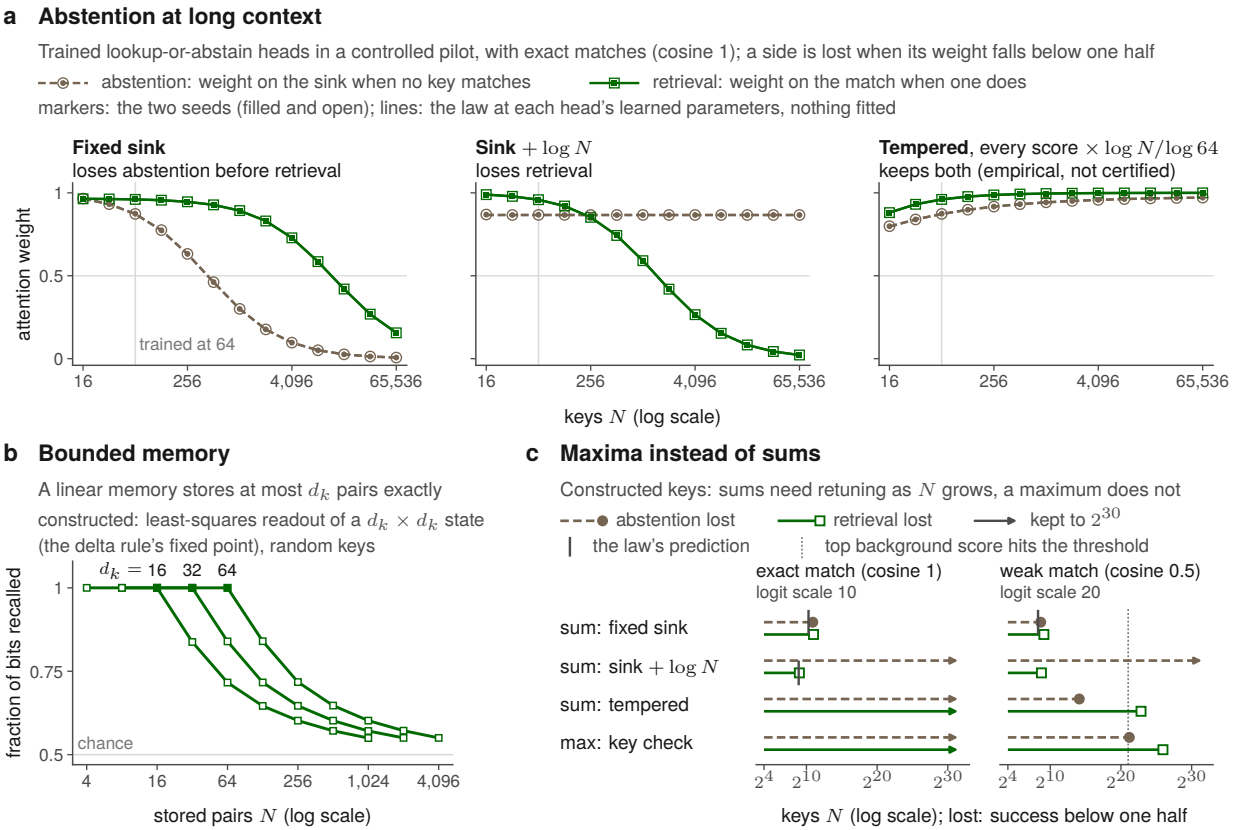


Figure 4: **With exact matches a fixed rule loses abstention or retrieval as N grows, while log-N tempering keeps both over the tested range (a) and a decision on the largest score needs no length schedule (c); a linear memory stores at most d_k pairs exactly (b).** (a) A controlled pilot (M§17.2), logit scale at most 10: means over about 512 queries per point (half of the 1,024 evaluated); the two seeds within 0.002 of each other. The curves plot sink and match weights: the fixed sink follows its weight law within 0.002 in each seed, and its retrieval accuracy stays at 1 even as its match weight falls. Sink $+\log N$'s retrieval accuracy falls below 0.9 from 2,048 keys in both seeds, as does sigmoid $-\log N$'s (not drawn). The tempered head reuses the trained fixed-sink head with its scores tempered at evaluation; its law is a stationary law, checked numerically, not proved. Three of the pilot's seven rules and one of four configurations are drawn. (b) Constructed memories (M§9.5): random unit keys and ± 1 values read by sign; 8 to 16,384 trials per point, no intervals saved. Exact recall of N arbitrary values needs $N \le d_k$ (M§18.3); the least-squares readout (the delta rule's fixed point) reaches it (filled: $N = d_k$) and, by construction, recalls the oldest pairs as well as the newest, whereas single-pass online writes (the delta rule, linear attention) are not exact even below $N = d_k$. (c) Constructed product keys (M§§9.3, 9.4), from historical records, recovered but not reproduced (Bruce, 2026e, Appendix C.2); sinks calibrated at 64 keys; 6,000 queries per point (three seeds pooled, no spread kept); sizes four-fold apart, so a loss (where success first falls below one half, interpolated in $\log N$) is located to within a factor of 4; two of four configurations drawn. The key check abstains unless its best score clears a threshold (M§8.4). Each predicted loss (vertical ticks) lies inside its measured four-fold step: the fixed sink's abstention in all four configurations, and sink $+\log N$'s exact-match retrieval, predicted at about 480 keys and lost between 256 and 1,024. The fixed sink's abstention and retrieval losses fall within one step here; at logit scale 20 with an exact match (not drawn) abstention is lost first, as in (a), at least six doublings before retrieval. The gap depends on the sink's place between background and match. With a weak match the key check outlasts tempering, losing abstention where the top background score hits its threshold. In trained 125M language models, log-N tempering did not keep abstention on top of YaRN, which already carries a log-length temperature (M§17.4).

every logit by $\log N / \log N_{\text{train}}$ on top of YaRN, which already carries a log-length temperature, did not keep abstention, so its lead prediction failed. Every rule that gives attention a null removed the first-token sink and most massive activations, as that test's registration predicted for language models and as earlier reports found; its registered counterpart in vision transformers did not hold, since the activation ratio rose in one seed. The others are pending.

4 What the question uncovered

Several of the program's main results carry the argument of Section 3: the six conditions, compensation, the study of explicit knobs, corners and the room. That argument is not repeated here, though some of the results it uses, such as the laws behind the room, reappear below. This section collects, in six themes, results that stand on their own for readers who do not care why transformers persist. Table 3 lists each with its strongest number, its evidence label and its chapter in the monograph.

4.1 Attention has an arrow, and default statistics of a single operator cannot see it

Many tasks need information to move one way along a sequence, and this theme asks whether attention has such a direction and whether the usual measurements can see it. Read as a Markov kernel, a head is the step rule of a random walk that moves from token to token with the head's attention weights as probabilities. Its stationary law is the long-run share of the walk's visits that each token receives, and every unmasked softmax head has a positive one. Every head with a positive stationary law splits exactly into three parts, orthogonal in the inner product weighted by that law: a shared preference for where attention gathers in the long run, a reciprocal exchange that flows equally both ways, and a circulation, a one-way current around loops of tokens that carries the head's direction (M§3.2). At the level of pre-softmax scores, every asymmetry between token pairs splits into a key-bias potential (a score offset per key, which is what an attention sink is) plus a doubly centred cycle (with zero row and column sums), and only the cycle circulates (M§3.3); on the complete graph of tokens this is the combinatorial Hodge decomposition used in statistical ranking (Jiang et al., 2011). This decomposition is proved, and a census applies it to 184,320 head kernels (certified to have an accurate stationary law and time reversal) in five encoders. Learned direction is specific and consequential. The census finds that circulation peaks in the middle of the network and that the positional heads of the language encoders, which put most of their attention on the previous or the next token, circulate almost maximally. The links that touch special tokens carry a share of the circulating current 9 to 24 times their share of all links (medians over layers 1 to 10) (M§10.3). In three vision transformers, reversing the learned circulation in every head of a middle layer changes predictions 8 to 12 times more than reversing feasible random circulations of the same size (alternatives that keep each head's stationary law and exchange fixed), as a ratio of mean changes. The prediction change from the learned reversal exceeds those circulations' mean change on at least 497 of 512 held-out images per model. An unfitted gradient readout (pairing the layer's loss gradient with the circulation update) predicts the signed change in loss under reversal with a pooled R^2 of 0.990–0.993, though its median per-image error at half dose (every head's circulation halved, once in its learned orientation and once reversed) is 8–16% (M§10.4). The learned orientation matters, though it is not always beneficial: reversing it can lower the loss, especially in the final layer.

Default diagnostics cannot see this direction. Time reversal, which runs the head's walk backwards, flips the circulation. It leaves unchanged, though, every stationary-weighted quadratic readout of one amplified operator (an averaging filter of a single operator): a sum over tokens of a signal times its own filtered version, with each token weighted by its long-run share. Such a readout is even in the circulation: when circulation is added, scalar summaries such as collapse move only at second order, while eigenvectors move at first (M§3.5). On per-layer lazy operators of BERT and DeiT (each head-averaged layer averaged with the identity), collapse readouts move by a median of 0.17–1.5% from their circulation-free values while eigenvectors turn by a median of 16–17 degrees (M§3.6). Time reversal also leaves these readouts unchanged across layers that share one stationary law and form a palindromic word, a sequence that reads the same forwards and backwards. A stack of distinct layers is not a palindrome, and there collapse readouts generally move at first order too. The principle reaches beyond attention. A statistic invariant under a natural involution (an operation that, applied twice, returns what it started from), such as reversal or the exchange of two classes, can see at most

the amount of the part that the involution flips, never its sign. Instruments that can see the sign must therefore be built on purpose. An associated paper on attribution gives a second instance, for the contrast between classes (Bruce, 2026f).

4.2 Depth is a distribution

What survives many layers of averaging depends not on the number of layers but on how often the signal takes a short route. Read a design as a path law over one fixed averaging operator: the distribution of how many averaging steps a signal takes. A mode, a pattern that one averaging step shrinks by a factor $0 < \mu < 1$ (its eigenvalue), keeps at least a fixed positive fraction of itself at every depth if and only if a fixed fraction of the paths stays within a bounded window of path lengths. Two designs with the same average depth can therefore meet opposite fates. This law is proved, and it covers oversmoothing in graph networks, the decay of label propagation and the rank collapse of pure attention repeated with one fixed operator (M§§2.2, 2.3, 4.1). For a stack of independent layers, how fast a mode is lost is set by the mixing budget, the expected number of layers in which the signal is averaged rather than passed on. With per-layer gains (the fraction of the mode each layer keeps) bounded away from zero, a bounded budget keeps a positive floor on what survives; the loss of the mode is polynomial when the budget grows logarithmically, stretched-exponential when it grows as a power of depth below one, and exponential when it grows linearly. To keep a floor, coherent updates (with a same-sign drift) therefore need residual branch scales of $1/L$ and incoherent ones (centered in expectation) $1/\sqrt{L}$, or smaller, with L the total depth. In the mean-field model, which replaces each token's own normalization by its average, pre-normalized transformers start training exactly on the critical line, the logarithmic-budget boundary where loss through depth is polynomial. On this line, random, untrained networks with bidirectional attention match the predicted decay exponent at width 512, with a largest deviation of 0.041 (M§§2.4, 2.6).

Trained language models appear to keep their mixing budget small. Across sixteen pretrained models, on 256-token inputs, the content-mixing budget, which counts only attention that moves information to another token, not to the sink or to the token itself, stays between 1.08 and 2.81 even at 48 layers. The family slopes of budget against depth on logarithmic axes are 0.075 for Pythia, 0.376 for GPT-2 and 0.673 for OPT under norm-based shares, which set each layer's mixing share by the length of its attention update against the length of the residual stream it is added to (under energy-based shares, which compare squared lengths, Pythia and GPT-2 do not grow and OPT's slope is near one). Deeper Pythia models route their surplus attention into the sink. This is partial evidence, because the model-level intervals are wide and depth and width grow together across each family (M§2.7).

Attention rollout (Abnar & Zuidema, 2020), a common way to trace attention through layers by multiplying residual-weighted attention matrices, turns out to have an exact meaning as a path walk. First-token collapse places all rollout mass on the first token, the causal sink. For any causal stack of stochastic attention, the largest row-wise ℓ_1 distance of rollout from this collapse is twice the worst-case probability that a walk back through the layers avoids the first token, where at each layer the walk stays put with probability equal to the residual share and otherwise steps by that layer's attention. The usual ways of handling sinks provably order the weight each content token (each token other than the sink) keeps on itself. A content token keeps the most weight on itself when the sink is treated as a skip, returning sink-directed mass to the token itself; the next most with a zero-value sink, which keeps the attention mass but writes nothing; and the least under eviction, which removes the sink and renormalizes attention over the remaining tokens. The order holds in all sixteen language models measured (M§§2.7, 8.3).

4.3 Averaging can amplify, and settling is not solving

Some models run one layer again and again on its own output, as looped and equilibrium models, iterative denoisers and test-time mixing of examples do, and their averaging rebuilds its weights from its current output at every step. Such adaptive averaging has a sharp stability test. Consider the anchored iteration $X_{t+1} = \alpha P(X_t)X_t + \beta A$ with $\alpha + \beta = 1$, which mixes the current states with weights rebuilt by the averaging rule P and pulls them back toward the fixed anchors at each step. Suppose the anchors and starting state lie in a ball of radius R, on which the rule changes by at most L_P per unit change of the state. Then a

positive stability margin $m = \beta - \alpha R L_P > 0$, where the pull back to the anchors outweighs the feedback from rebuilding the weights, certifies that the ball is invariant and the map a contraction, so the iteration converges to a unique fixed point in that ball by the contraction-mapping theorem (Banach, 1922). This threshold cannot be improved over the class of such averaging rules. Tied dot-product softmax at logit scale s has $L_P \le 2sR$; the proof is formalized in Lean (M§7.1).

Settling, though, does not mean behaving. Every equilibrium is a positive average of the anchors, which cannot amplify when its weights are held fixed (Section 3.1); yet because perturbing the input anchors also shifts the equilibrium weights, the equilibrium's sensitivity to input changes can approach β/m, the largest sensitivity that the stability margin m allows. Ordinary softmax kernels settle while amplifying: a Euclidean response of 18.875 falls to 1.766 when the equilibrium weights are frozen instead of adapting to the perturbation. A fixed average's response can exceed one in the Euclidean norm, but a fixed average never enlarges the largest change at any one token; adapting weights can. A photographic denoiser amplifies a selected perturbation 19.12-fold. Test-time mixing blends each test image's frozen features with those of the other images in a context of 32 randomly drawn CIFAR-10 images before a fixed classifier reads them, much as transductive few-shot methods mix test images. Run to convergence, it cuts accuracy from 88.09% to 35.87% on test images that have the same companions under every method, whereas the same mixing with its weights frozen at the unmixed input keeps 88.02% (M§§12.1–12.4). These maxima were selected from grids: they show what can happen, not how often it does.

In test-time mixing, as in any row-stochastic context mixing started at the input and followed by an affine readout, each prediction's margin between two classes splits exactly into the sample's own margin and its companions' margins for the same two classes, weighted by the mixing. This identity holds at every iteration and at the fixed point. That makes it possible to see which companions overturned a correct prediction: when strong mixing drives each sample's weight on itself down near the anchor floor, the share of itself a sample keeps once mixing has made the weights uniform (the anchor's share plus an equal part of the rest), the companions' contributions outweigh its own, and a label-free correction that subtracts the context mean recovers 87.96% (M§§12.4–12.6). More generally, a computation that settles to machine precision certifies neither its response, nor the uniqueness of its endpoint, nor that the endpoint attracts, nor that a loop built on it is stable; each reading needs its own check. In a constructed two-token example, a float64 finite difference can read a response of 55.83 against a true 9,998, although the unperturbed run it starts from (the construction's clipped form) has settled to a residual of 4.97×10^{-13}. A root solver's success does not show that the root attracts nearby states. And in an outer loop, a larger loop that uses the settled result, uniformly random unit probes in n dimensions read an amplification G concentrated in one direction (rank one) as $G/\sqrt{n}$ in root mean square, so a directional probe can miss an expanding mode. These are proved and shown on constructions and small systems; trained looped and implicit models are still to be tested (M§§13.2–13.6). Finally, applying the same stability margin to training ties the logit scale to weight decay. Tied dot-product softmax on a domain of radius R has loop gain at most $2sR^2$. If decoupled weight decay λ is modelled as the anchor in a training step, the learning rate scales both the feedback and the anchor pull and so cancels, giving the sufficient cap $s_c = \lambda/(2R^2)$ (M§7.3). In the one test of this coupling so far, relaxing the anchor without any feedback reproduces all 170 flags raised by the test's divergence detector, with one false alarm among 550 cells. Because the detector fires on the anchor relaxation alone, its flags cannot test this coupling. The threshold is proved; the cap it implies is untested in language models (M§7.4).

4.4 Length laws: sums, maxima, leaks and capacity

A head should be able to deliver nothing when nothing in its context is relevant, which is abstention. Every patch that lets attention abstain is the same gate on an ordinary softmax: a zero-value sink or register, an off-by-one null, a learned sink logit or a headwise output gate. With bounded scores and one null, the option that absorbs attention and delivers nothing, shared by absent and present queries (queries without or with a matching key), none keeps abstention and retrieval as the context grows. Scaling every score with the logarithm of the length is one way to keep both when the score margins between match, null and background are strictly separated (Section 3.7; M§9.2).

In small byte-level language models trained without positions, a learned null, measured relative to the mean content logit (the mean score of the real keys, as against the null's), behaves as a fixed one: it does not rise

with the training length. In these models, the log-odds of abstaining fall close to the fixed-null law's rate, a fall of one unit each time the length grows by a factor of e (about 2.7), across training lengths and, for models trained on 128 tokens or more, with context length within each model (medians over heads and five seeds) (M§17.4).

This is one instance of a broader rule. With an exchangeable background, one whose scores have the same joint distribution in any order of the items, a decision that sums over all N items at a fixed logit scale must be retuned like $\ln N$, while a decision that reads a maximum need not be retuned against backgrounds bounded below its threshold (M§9.4). A companion bound on routing assumes independent background scores from one distribution, an independent target score, a uniformly placed target and a likelihood ratio rising with the score. Given the same scores, no router, whatever rule it uses to name an item, succeeds more often than picking the highest score. Past the evidence wall, the crowd size at which the best background key outscores the target half the time, only better keys help (M§9.3). Both bounds are proved, with synthetic evidence; that decisions on maxima escape the schedule in trained models is partial (Section 3.7).

A machine that reads a store written over time often needs the latest write, not just a matching one, so it must tell later writes from earlier ones. Soft routers, readers whose weights spread over every item as a softmax's do, pay for that order as well as for the size of the crowd. To order a pair of identical writes by a ratio $r > 1$ is to give the later write at least that multiple of the earlier write's weight. With a logit range B, a router that orders its memories through a positional logit term can order at most a fraction $K/(K+1)$ of pairs of identical writes among many positions, where $K = \lfloor B/\ln r \rfloor$; a staircase positional bias attains the bound. When every position must stay retrievable, a mean-field calculation, which replaces the crowd of competing keys by its average, shows that the positional terms are paid from the router's score margin above its gain wall (their exponential mean cannot exceed the exponential of that margin), the crowd size at which the competing keys would take half of its weight. Timestamps and overwriting give order without spending that margin (M§10.7). In the mean-field calculation of Bruce (2026e), at logit scale $\tau = 10$, with 1,024 keys in 128 dimensions and an ordering ratio of 9, the ceiling on the fraction of pairs ordered is 0.60, against 0.22 for the best linear bias.

When one context packs several documents, a head is often meant to keep each token's attention inside its own document's block. Small leaks across a block boundary grow with path length, while exact block zeros stop them. Consider a path-law filter built from one fixed averaging operator, each of whose rows leaks at most ϵ across a block boundary in one step: over a walk of random length k drawn independently of the steps, it carries at most $\epsilon\,\mathbb{E}[k]$ across that boundary. Exactly block-diagonal attention, by contrast, keeps every analytic filter block-diagonal at any depth. With logits bounded by a fixed constant, dense softmax cannot make its leak vanish as the context grows: its smallest possible leak is the routing budget's $\log N$ growth seen from the other side: holding the weight on the other blocks down as they grow takes a score range that grows like $\log N$ (M§§6.4, 9.1). These laws are proved, with evidence on constructed operators.

A fixed-size memory's recall limit is a capacity, not a distance: however far back the writes are, a linear memory can recall every assignment of values to N keys exactly if and only if the keys have rank N. Writes that contract the state by a common factor (a strict decay) bound the state at every length when keys and values are bounded; non-expansiveness alone does not give that bound (M§§18.2, 18.3).

4.5 Protections the transformer already has

Several proposals add explicitly what the transformer already has, such as signed attention for subtraction. Two stochastic heads, taken as arbitrary patterns of positive weights, can represent any signed kernel with constant row sums, so multi-head attention with signed output maps can subtract without signed attention weights; real heads, whose scores come from low-rank queries and keys, may not reach every pattern. In a single small block trained for 500 steps, difference retrieval (recovering the difference of two values) under softmax attention reaches a mean R^2 of 0.46 with one head and 0.998 with four (25 matched runs at each head count). In separate six-block language and vision models, learned signed gates set the weight of a signed branch beside ordinary softmax; those weights average below one half, on the softmax side, in every run, though their final values depend on initialization: they start at 0.119 or 0.5, none of the paired starts met the registered convergence tolerance, and the language model's softmax baselines mostly missed their task

threshold (M§15.5). Positivity of the attention weights keeps outputs in the hull of the values, but it does not cap the spectral norm at one, so subtraction built from heads still needs its norm controlled. This is partial evidence: a sweep over head counts at larger scale is pending.

Invertible is not the same as isometric. An invertible block allows its input to be recovered exactly from its output, which still lets the block write features; an isometric block preserves distances between inputs, and a norm-preserving block keeps the length of each state. An invertible block built from additive reversible coupling matches a pre-normalized transformer on dictionary retrieval (mean $R^2 \geq 0.99996$ over five seeds and 8–64 associations). On CIFAR-100, it sits a mean 0.78 percentage points below that transformer (95% interval $[-1.98, 0.43]$, five paired seeds) with a fifth of the parameters. At depth 48, it needs 0.24 times the extra training memory of checkpointing, gradient buffers included. By contrast, input-conditioned orthogonal transport reaches only 0.18–0.26 on retrieval (mean R^2 at 16 associations, on a synthetic bidirectional task), and its token-rotation arm, which preserves state norms and the channel Gram matrix, loses 10 points on CIFAR-100 (M§§11.2–11.5). These observations concern that transport family; the arms were not parameter-matched, and the implementations' qualification is incomplete (their reference-parity checks could not run), so this too is partial.

Production vision transformers grow sinks of their own. Six of seven pretrained ViTs grow a population of stationary-mass outliers (patch tokens that draw an outlier share of the attention chain's stationary mass) that first appear in at least half the sampled images between layers 6 and 11. These sinks do not have zero values in the six checkpoints tested for it: zeroing their values still moves the output logits, so they carry content of their own. Whether the same tokens remain sinks across layers depends on the checkpoint (M§8.5). The census is observational: training recipe, data and scale are confounded across the checkpoints.

4.6 How to know what you know

A strong signal need not be useful. On fifteen neighbour graphs of frozen CIFAR-100 features (three backbones, five seeds each), each held bit-identical while the labelled set is cut from 40 to 4 examples per class, the worst tested accuracy effect of propagating predictions over the graph turns from harmful into helpful. The paired swing is 10.94 points (95% interval [10.23, 11.64]); the registered primary, pooling 4 and 8 per class, is 8.78 [8.15, 9.41]. The best spectral gauge of signal, a score computed from the graph's spectrum and the labelled nodes, rises with label density. It therefore approves propagation in exactly the high-density cases, where the worst tested effect of propagating is negative and the classifier already has little headroom, little that propagation could still add, and rejects it in the low-density ones, where that effect is positive. Utility is signal times headroom, though the label-density sweep does not separate headroom from the coverage of nodes with no labelled neighbour (M§4.4).

Other results. A few more results across the section's themes are partial, synthetic or still unrun, and are listed here to complete the inventory. Two training loops that share state can each pass a single-loop certificate while the pair diverges. On a trained self-distillation stack, where a student and a moving-average teacher form such a pair, a single composite stability margin combining self-distillation with the labelled task's anchoring curvature did not certify representation collapse; a two-loop eigenvalue pencil checks their joint stability through a matrix's eigenvalues, certifying the pair and naming the coupling to cut, so far only on synthetic rigs (M§13.8). At score collapse, when attention keys become near-equal, a fixed null competes with all of them, so only a null coupled to the temperature, or gated by the query, should repair abstention across temperatures with one shared setting, a prediction not yet run (M§8.6). In graph-consistency regularization, holding the perturbation and the loss fixed, the neighbour graph is what carries the regularizer's improvement, on one backbone architecture so far (unreleased work). The anchored causal resolvent of a frozen, factorized kernel, which sums causal walks with a restart at the input, computes in one causal pass with a state independent of length, a design without a trained test (M§18.1). And when attention is block-diagonal, its blocks need not share one filter: exact block zeros let each block carry its own filter with no contamination between blocks, shown on constructed operators (M§6.4).

Table 3: What the question uncovered: the headline results by theme, with the strongest number, the evidence label and the chapter of the monograph (Bruce, 2026a). A dagger marks an exact identity: it holds by construction for every operator of its class, and its value is what it makes measurable.

Result	Strongest number	Label	Chapter
Attention has an arrow, and default statistics of a single operator cannot see it			
Every unmasked attention head splits into preference, exchange and circulation	Applied to 184,320 certified kernels	proved†	3, 10
Trained attention's direction is specific, and it matters	Reversal: 8–12 times the matched controls' mean change; above their mean on $\geq$ 497 of 512 images	measured	10
Standard diagnostics of a single operator cannot see which way attention points	Collapse readouts move a median 0.17–1.5% while eigenvectors turn a median 16–17°	proved	3, 10
Depth is a distribution			
Depth is a distribution: short paths decide what survives	Retention bounds checked on 2,000 random path laws, largest violation 1.1×10^{-16}	proved	2
The expected number of mixing layers sets how fast depth forgets	Random Pre-LN transformers with bidirectional attention at initialization match the predicted exponent with a largest deviation of 0.041	proved	2
Deeper trained language models send their surplus attention to the sink	Mixing-budget slopes 0.07 (Pythia), 0.38 (GPT-2), 0.67 (OPT); wide intervals	partial	2
Attention rollout has an exact probabilistic meaning	The sink-treatment order holds for every token in 16 LMs	proved†	8
Averaging can amplify, and settling is not solving			
A sharp, machine-checked stability test for adaptive averaging	Formalized in Lean: 56 audited declarations	proved	7
A settled average can amplify changes and destroy accuracy	Matched-query accuracy 88.09% $\rightarrow$ 35.87%, with frozen weights at 88.02%	measured	12
Each prediction's class margin splits into its own and its companions'	A label-free context-mean correction recovers 87.96% (input 88.09%)	proved†	12
Converging is not the same as being right or stable	A finite difference on a settled solve reads 55.83 against a true 9,998 (unperturbed residual 4.97×10^{-13})	partial	13
One stability threshold links logit scale to weight decay	$s_c = \lambda/(2R^2)$; a feedback-free relaxation reproduces all 170 detector flags, with one false alarm in 550 cells	proved	7
Length laws: sums, maxima, leaks and capacity			
Softmax's $\log N$ law: abstention and retrieval cannot both be fixed	Fixed sink 0.874 $\rightarrow$ 0.007 at 65,536 keys, within 0.002 per seed	proved	9, 17
Sums need a length schedule; maxima do not	The fixed softmax sink fails at the predicted onset in 4 of 4 constructed configurations	partial	8, 9, 17
Soft routers pay for order out of their score range	Range B orders at most a fraction $K/(K+1)$ of the pairs of identical writes, $K = \lfloor B/\ln r \rfloor$; the mean-field ceiling is 0.60 at $\tau = 10$, against 0.22 for the best linear bias	proved	10
Small attention leaks can grow with depth; exact block zeros stop them	A 1% leak reaches 26.2% after 32 steps; exact zeros stay zero	proved	6
A fixed memory's recall is limited by capacity, however far back	Exact linear recall of arbitrary values needs $N \leq d_k$ per head; a trained memory holds, on 32 pairs, 0.997–0.999 across an 8,192-token gap	proved	18
Protections the transformer already has			
Signed attention is already inside your transformer	Difference retrieval R^2 0.46 with one head, 0.998 with four	partial	15
Invertible blocks keep capability; the tested orthogonal transport loses it	Retrieval $R^2 \geq 0.99996$ for additive coupling vs 0.18–0.26 for orthogonal transport	partial	11
Production vision transformers grow sinks that are not empty	Six of seven ViTs grow sinks, born at layers 6–11; not value-null	measured	8
How to know what you know			
A strong signal is not the same as a useful one	A paired swing in propagation's accuracy effect of +10.94 pp [10.23, 11.64] from 40 to 4 labels per class, on 15 graphs each held bit-identical across label densities (registered primary, pooling 4 and 8: +8.78 [8.15, 9.41])	measured	4

5 What holds in trained networks, and what is shown in idealized settings

Many results in the collection are proved for one operator applied again and again: a fixed matrix raised to powers, or one adaptive map iterated to its fixed point. A trained transformer is not that object. It applies a different, input-dependent operator at every layer, with residual paths, normalization and an MLP between, and it applies each layer once. So each claim this paper quotes answers a second question beside the strength of its evidence: does the statement hold for the networks people train, or does it describe an idealized object whose behaviour trained networks may or may not share? There are five answers: a statement may hold across its stated class, hold under premises checked on the network, be shown in an idealized setting, be measured in trained networks, or be an interpretation with nothing to transfer, and Table 4 gives each claim's answer (along with the claims still pending).

Exact for every network of the stated class. Some statements are identities that hold at every layer and for every input, so they apply to trained networks as they stand. Every softmax head is a row-stochastic kernel whose outputs are convex combinations of its values, and every zero-valued null patch (such as a zero-value sink, register or learned sink logit) and headwise output gate factors as a gate times a softmax (Section 3.2) (M§8.2). Every head with a positive stationary law splits exactly into preference, exchange and circulation (Section 4.1) (M§3.2). A mixer that only averages keeps every output in the hull of its inputs and cannot raise their rank (Section 3.2) (M§11.1). After row-stochastic context mixing and an affine readout, each prediction's class margin splits exactly into its own margin and its companions' margins for the same two classes, weighted by the mixing (Section 4.3) (M§12.5). One result of this kind is about whole stacks: attention rollout's distance from collapse onto the first token is twice the worst-case survival probability, the probability that a walk back through the layers avoids the first token (Section 4.2). This holds for any ordered product of causal stochastic kernels in which the first token is absorbing, with per-token residual shares, whatever the layers (M§8.3). It is exact for that linearization of attention flow, not for the full block.

Exact under premises checked network by network. Other statements are theorems whose conditions a given network either meets or does not, and the conditions can be measured on it. The routing budget, and the two-sided law for a null that absent and present queries share (Sections 3.2 and 3.7), hold for any softmax row whose content logits lie in $[-\tau, \tau]$, with τ read off the network (M§9.1). The anchor exception (Section 3.4), which rules out compensating for a missing anchor, holds for any stack meeting these conditions: the layer maps restrict to a common discriminative invariant subspace with norms at most one; the contraction product across layers tends to zero; and the recovery map's Lipschitz gain is uniformly bounded. A version based on Dobrushin's coefficient, the factor by which one averaging step is guaranteed to shrink the spread between tokens, covers sink-like attention with shared feature maps (M§5.5). Recall capacity (Section 3.7) holds for linear read-back from a memory, through the rank of its keys (M§18.3). The parity law for stationary-weighted quadratic readouts (Section 4.1), which leaves them unchanged under time reversal, holds for one amplified operator with a positive stationary law and for palindromic words of layers (layer sequences that read the same forwards and backwards) that share one; a stack of distinct layers is not a palindrome, and there the readouts generally move at first order in the circulation (M§3.5). The certificate for adaptive averaging (Sections 3.2 and 4.3) holds for any anchored iteration whose stability margin m between the anchor and the feedback is positive (M§7.1).

Where the idealization meets practice. That last statement is the clearest case of the difference between one operator applied over and over and a different operator at each layer. The certificate, and with it the gain-control condition and the amplification results of Section 4.3, concern one adaptive map applied repeatedly. Weight-tied looped models, deep equilibrium models, iterative denoisers and test-time context mixing reuse the same parameters at every step. Their averaging therefore belongs to that class when it takes the anchored form and meets its premises (M§4.2). At a fixed point, freezing the attention weights gives one fixed operator, the object the static theory describes. Linearizing the adaptive step around that fixed point adds a second term, the feedback from rebuilding the weights when the input changes, which is how a settled average can still amplify perturbations (M§§1.4, 12.1). Away from a fixed point even these models apply a different operator at each step, because attention depends on its input. A standard stack applies each of its distinct layers once, so it is outside the class. The per-layer feedback-gain bound $\kappa_f \le 2sR^2$, in terms of the

logit scale and the representation radius (Section 3.2), still holds for each of its layers with tied dot-product scores, and with the projections' norms as extra factors for projected scores. But contraction and a unique attracting state are not statements about it.

Shown in an idealized setting. A third group is proved for an idealized object, and its relation to a trained network is a question rather than a theorem. The anchor's if-and-only-if condition for keeping a mode through depth (Section 3.2), depth as a distribution over path lengths (Section 4.2), and the growth of cross-block leaks with path length (Section 4.4) are proved for path laws over powers of one fixed operator (M§§2.1–2.3, 6.4). Linear propagation over a fixed graph satisfies this exactly; a transformer does not. What carries over is the qualitative lesson of these results. Bounds on realized products of distinct mixing kernels hold however the kernels arose (M§2.5). The Dobrushin bound covers attention that changes from layer to layer while it mixes strictly, every layer's mixing step shrinking the spread between tokens by at least a fixed factor, with shared feature maps (M§5.5). So averaging without short paths forgets in trained stacks too, though these bounds do not model the full block. What does not carry over is the exact criterion and its rates. The mixing-budget law (Section 4.2) is stated for the expected energies of residual stacks with random feature maps and mean-field normalization, and it is matched by random, untrained transformers with bidirectional attention at initialization (M§§2.4, 2.6). For independent or exchangeable backgrounds on constructed keys, it is shown that routing decisions based on sums over all competitors at a fixed logit scale need a $\log N$ schedule while decisions based on maxima need none against backgrounds bounded below their threshold (Sections 3.7 and 4.4) (M§9.4). The mean-field law that the lookup-or-abstain pilot of Section 3.7 tracks assumes an independent background of fixed, isotropic keys (M§§9.2, 17.2).

Measured in trained networks. The fourth group is measured on trained networks directly, and its reach is the reach of the models measured. Removal signatures, compensation and the study of explicit knobs (Sections 3.2 and 3.4) come from controlled runs, mostly below 25 million parameters (M§§5.2, 15.1, 15.4). The circulation census (Section 4.1) covers 184,320 head kernels of five trained encoders, with reversals tested in three vision transformers (M§§10.3, 10.4). Depth and sinks (Sections 4.2 and 4.5) were measured on sixteen pretrained language models and seven production vision transformers (M§§2.7, 8.5). For long-context abstention (Section 3.7), the prespecified sink test on a production language model, gpt-oss, was undecided by its rule, though descriptively its slopes lean toward compensation through position (M§17.3). Controlled causal language models at 125 million parameters were trained at 1,024 tokens and tested to 8,192, with three seeds per rule. In these tests, log-N tempering on top of YaRN, which already carries a log-length temperature, did not keep abstention and raised long-context perplexity. Every tested rule that gives attention a null removed the first-token sink (M§17.4). The 350-million-parameter leg is not yet measured (Appendix B.4).

Interpretations. The sweet spot, the price list, the retrodiction and the successor checklist (Sections 3.3, 3.6 and 3.7) are readings under a model, not statements about networks, so there is nothing for them to transfer. They stand or fall with the predictions stated in advance (Section 6.2). The final group of Table 4 lists the claims whose test or theory is still pending, with nothing yet to transfer.

How to read the rest of the paper. Where Sections 3 and 4 state a law for one operator, they say so, and the practical rules in Section 6 carry those scopes forward. Table 4 is the place to check what a statement covers before applying it to a network.

Table 4: What holds in trained networks, claim by claim. Each claim of Sections 3 and 4 is grouped by its transfer label; beside it, what has been measured in trained networks, with the models and the scale, or the setting the statement covers when nothing has been measured. Each label's note is in the claims ledger (Appendix C). A dagger marks an exact identity: it holds by construction for every operator of its class, and its value is what it makes measurable.

Claim	In trained networks, or the setting covered
Exact for every network of the stated class	
A mixing layer is an averaging operator rebuilt from its input	Every softmax head, at every layer and input, is a row-stochastic kernel $P(X)$ whose outputs are convex combinations of its values. The path-lottery form $\mathbb{E}[P^K]$ describes powers of one operator; a trained stack is an ordered product of different $P(X)$, with MLPs between.

Claim	In trained networks, or the setting covered
Attention cannot abstain unless it is given a way out	Sigmoid attention in small ViTs (3 seeds); an output gate in a small causal LM (3 seeds; read from value norms, as the stationary-mass sink test is vacuous under a causal mask); 2,700 operators captured from seven public LMs at 64 tokens.
A head that cannot circulate cannot shift tokens around a cycle	Tied (Gram) score models reach 60% and 76% successor retrieval at depth 2 and 4; under a causal mask, with both arms masked, they copy less surely at varying offsets (mean per-seed cross-entropy ratio 3.4, accuracy 0.8 pp lower; 5 seeds); reversing learned circulation in trained retrieval models turns successor into predecessor retrieval.
Moving information around is not enough; something must write	Transport-only mixers trained on retrieval and CIFAR-100 (5 seeds; arms not parameter-matched).
Every unmasked attention head splits into preference, exchange and circulation†	Applied to 184,320 certified kernels of five trained encoders.
Attention rollout has an exact probabilistic meaning†	The ordering holds for every token in sixteen pretrained language models.
Each prediction's class margin splits into its own and its companions'†	23,040 endpoints reproduced on test-time mixing over real image contexts.
Converging is not the same as being right or stable	Constructions and small systems; trained looped and implicit models are pending.
Soft routers pay for order out of their score range	A combinatorial bound for any soft router whose positional logit term has range B; its cost to the score margin, when every position must stay retrievable, is a mean-field statement.
Signed attention is already inside your transformer	Small trained models (5 seeds); a head-count sweep at 125M–350M is pending.
Exact under premises that are checked network by network	
Routing and content need separate maps	Tied value maps at head dimension d_b in small trained models; the tying tax at depth 24 on CIFAR-100 (5 seeds).
Self-reinforcing weights need their feedback held below the anchor	Removal signature synthetic only.
One key among N needs a score margin growing like $\log N$	Accuracy onsets of small trained retrieval heads (15 cells); a training-free $\log N$ re-score on trained ViTs, on upsampled images with interpolated positions; the background's growth with position in trained causal language models at 0.6M and 125M parameters.
Every machine pays to name one of N, in some currency	All tests synthetic; no trained language model yet.
No capacity of bounded gain buys back short paths	Identity removal in trained small ViTs; the capacity sweep in trained networks is pending.
Under concave prices, the best design is a corner	Deployed hybrid fractions, recorded descriptively for two models.
No fixed rule keeps both abstention and retrieval at long context	A controlled pilot of trained lookup-or-abstain heads (fixed isotropic keys; 2 seeds, 4 configurations); the gpt-oss test was undecided by its prespecified rule (descriptively, toward compensation); in 125M language models (3 seeds) log-N tempering did not keep abstention on top of YaRN, and helped beyond the training length on interpolated positions without YaRN's temperature.
Bounded memory buys length robustness and pays in recall	A gated delta rule on multi-query associative recall; a decay gate in a trained hybrid (five seeds per variant).
Standard diagnostics of a single operator cannot see which way attention points	Native-flow readouts of lazy, head-averaged operators from two small trained encoders (BERT, DeiT).
A sharp, machine-checked stability test for adaptive averaging	A real photographic denoiser; test-time mixing over real image contexts.
One stability threshold links logit scale to weight decay	The divergence detector's anchor-relaxation confound is measured on 550 cells (all 170 flags reproduced, one false alarm); the cap is untested in language models.
Softmax's $\log N$ law: abstention and retrieval cannot both be fixed	A controlled pilot of trained heads (2 seeds; fixed isotropic keys); small byte-level LMs without positions follow the stationary law within each model trained on 128 tokens or more; the gpt-oss test leaned toward compensation without settling it; in 125M LMs with rotary positions, abstention stays nearly level inside the training length through position.
A fixed memory's recall is limited by capacity, however far back	A gated delta rule on MQAR.

Claim	In trained networks, or the setting covered
Shown in an idealized setting	
The same laws govern graphs, denoisers and attention	The realized filters on a 1,500-node connected subgraph of a citation graph, which match the closed form to 1.5×10^{-14} (the collapse-ratio identity holds to 6.5×10^{-16}); a real photographic denoiser; not a trained transformer's forward pass.
Deep averaging forgets unless some paths stay short	Removal measured in trained small ViTs (identity path removed: chance at depth ≥ 12; Post-LN cliff at depth ≥ 18 on CIFAR-10) and in weight-tied looped blocks (96 loops, 3 seeds); all below 25M parameters.
Decisions on maxima need no length schedule	Small trained heads on constructed geometries (3 and 5 seeds); in 125M language models (3 seeds), with YaRN, a head built on maxima was the only rule whose null and match weights (attention weights, not behaviour) both fell by less than half a nat, on the seed means, not in every seed; it failed natively.
Depth is a distribution: short paths decide what survives	Graph filters on real graphs.
The expected number of mixing layers sets how fast depth forgets	Matched by random, untrained transformers with bidirectional attention at initialization (width 512, largest exponent deviation 0.041).
Sums need a length schedule; maxima do not	Constructed product keys (3 seeds pooled) and small trained heads on constructed geometries (3 and 5 seeds).
Small attention leaks can grow with depth; exact block zeros stop them	Constructed operators; trained and pretrained tests are pending.
Measured in trained networks	
Trained networks buy back what a component lacks	Small trained models below 25M parameters: a rank-$(C-2)$ content map for $C = 10$ classes trained with its readout (99.84% accuracy), tied value maps, multi-head subtraction, and the tying tax at depth 24 on CIFAR-100 (5 seeds varying only the initialization).
Making a protection explicit does not beat a block that has it	Small models, mostly below 25M parameters; several planned tests remain.
A 22M-parameter accuracy margin has not yet been replicated	The minimal block's margin over ViT-Small on Tiny-ImageNet and ImageNet-100, 22M parameters, 5 seeds.
Trained attention's direction is specific, and it matters	A census of five trained encoders; reversal tested in three ViTs, one checkpoint each, on at least 497 of 512 images per model.
Deeper trained language models send their surplus attention to the sink	Sixteen pretrained language models (GPT-2, Pythia, OPT); model-level intervals are wide.
A settled average can amplify changes and destroy accuracy	Real images and frozen features; not a trained transformer's forward pass.
Invertible blocks keep capability; the tested orthogonal transport loses it	Retrieval and CIFAR-100, 5 seeds; arms not parameter-matched.
Production vision transformers grow sinks that are not empty	Seven pretrained ViTs, one checkpoint per setting; recipe, data and scale confounded.
A strong signal is not the same as a useful one	CIFAR-100 kNN graphs from three backbones, 5 seeds each; a second dataset is pending.
Interpretations	
We read the standard block as paying for every condition at once	Tuned standard components carry the protections in small trained models (Pre-LN anchor 0.998; Switch balancing; multi-head subtraction).
Hardware sets the prices; the conditions are what is priced	A reading that joins the hardware account to the conditions; there is nothing to transfer.
Relaxing a constraint buys a capability and obliges a re-supply	A reading of the lattice; the individual re-supply obligations are proved in their chapters where stated.
Read backward, adopted changes re-supply the same protections	A reading of adopted changes; there is nothing to transfer.
Read forward, the conditions rebuild a transformer	The minimal block in small trained models: detailed balance to 10^{-12} in 50 encoders; 20/20 paired wins over ViT-Tiny on CIFAR-10/100.
What any successor must still supply	A checklist read from the stated classes; there is nothing to transfer.
Pending	
Deficits may cost less as models grow	A hypothesis about trained networks at scale; its test is pending.
What makes a good writer is still open	Open: there is no theory yet to transfer.
Predictions stated before the evidence	Predictions about trained networks, stated before their tests; the tests already run are reported with their outcomes.

6 What this changes

6.1 For architecture work

The account gives a way to read an architecture proposal. Ask which constraint of the block it relaxes, what that relaxation buys, and how it re-supplies what the constraint guaranteed (Section 3.3); a proposal that relaxes a constraint and re-supplies nothing has a failure mode waiting for it. Expect a successor to win on a resource axis, such as memory, length or speed, at matched quality, rather than on quality at matched cost. Read small-scale wins with compensation in mind: if spare capacity makes up for weakened components, an improvement measured in a small model may shrink in a large one, although that is still a hypothesis (Section 3.4). And test a protection by removing it and looking for its removal signature, not by adding an explicit knob for it, because a tuned block already supplies the protection (Section 3.4).

Some rules can be used now, each with the evidence its chapter gives. Give attention a way to deliver nothing, such as an output gate on each head or a null, an option that absorbs attention without delivering anything, like a learned sink (Section 3.7): in trained 125M language models both remove the first-token sink and most massive activations at no cost in perplexity at the training length (M§17.4).

When extending a model beyond its training length, keep its positions in distribution with interpolation such as YaRN (Section 3.7). Without interpolation, ways to deliver nothing that act outside the attention scores, such as an output gate, degrade about as fast as plain softmax, while a fixed null in the scores, such as a sink logit, breaks down. Scaling every logit by $\log N / \log N_{\text{train}}$, the ratio of log context length to log training length, helped where positions were interpolated, in vision transformers at higher resolution (on upsampled images a constant temperature fitted per resolution did as well) and in causal language models; on top of YaRN, which already carries a log-length temperature, that scaling hurt, and without interpolation YaRN's temperature alone hurt as well (M§§9.2, 17.4, 17.6).

With bounded keys and values, bound a recurrent state with decay that makes every write contract by a common factor, rather than by limiting the write step alone, which makes writes non-expansive without bounding the state (Section 3.7). In that setting, size each head's state to the associations it must read back exactly, at most d_k of them, the key dimension, for a linear memory (M§§18.2, 18.3).

Scale coherent residual branches (with a same-sign drift) by $1/L$ and incoherent ones (centered in expectation) by $1/\sqrt{L}$, with L the total depth (Section 4.2), as the fixed-operator energy model prescribes to keep a floor through depth (M§2.4).

In a decoder, mask any layer built from a time reversal or a circulation flow (Section 4.1) and test that it mixes no future tokens into earlier outputs, because the reversal or flow of a causal operator looks ahead to future tokens (M§10.8).

And read row-stochastic test-time context mixing with an affine readout through each prediction's own and companion margins, which split the output class margin into the sample's own contribution and the context's and show when companions overturn a correct prediction (Section 4.3) (M§12.5).

6.2 Predictions stated in advance

The account can be wrong, and the paper released with this one on the transformer as a constrained equilibrium of protections states the predictions whose failure would show it (M§20.1). In brief:

- The log-odds of sink mass in dense layers with learned sink logits fall with the logarithm of query position at a slope near -1, and banded layers stay flat beyond their window. The test on gpt-oss (Section 3.7) was undecided by its prespecified rule, though descriptively its slopes lean toward compensation through position, and in trained 125M models the fixed sinks took the compensated branch on text. Since the slope of the sink log-odds is minus the growth of each head's background, the rate at which the log of its background sum (the softmax's denominator without the sink's own term) rises with the log of the query's position, the prediction is about that growth: near one without positions, as in small byte-level models above the shortest tested training length; below one inside

the training length with rotary positions, as in 125M models; and lower in more local heads. A rotary head whose background grows at the full rate inside its training length, or a local head whose background grows faster than a global one's, would count against it. This form was fitted to those models, so only other models can test it.

- Any widely adopted attention whose rows of weights need not sum to one ships with norm control. Current designs already pair the two, so the prediction concerns future designs: one adopted widely with neither output nor logit normalization would count against it.
- Sigmoid attention with a $-\log N$ bias, its offset shared by absent and present queries, loses single-key retrieval weight like $1/N$ unless its match logits grow like $\log N$. In trained 125M models with YaRN, its match weight fell from the training length to eight times it in all three seeds, by 1.14 in log weight on average against the prespecified minimum drop of 1.0 (one seed fell by only 0.48), about half the $\ln 8 \approx 2.08$ of a strict $1/N$ law; no head there met the retrieval threshold, so this concerns attention weights. The same registered prediction expected the length-calibrated sink to lose its match weight; with YaRN it lost its null mass instead (M§17.4).
- The improvements from signed attention shrink as heads are added at fixed width.
- Masked diffusion language models carry more delivered circulation (the part of a layer's attention output that the heads' one-way current writes through their values and output maps) in early layers than masked-language-model encoders of similar size.
- Any future statistic that predicts the sign of the effect of reversing a trained head's circulation changes under time reversal (Section 4.1).
- Self-referential logit instabilities, in which attention's weights feed back on the tokens that set its scores, appear where the feedback outweighs the anchor, beyond the certificate (Section 4.3); with weight decay as the anchor and tied dot-product scores, the certificate's threshold logit scale is $\lambda/(2R^2)$, set at a fixed radius R by the weight decay alone. Because the certificate is only sufficient, this is a weak test: an onset of logit growth that moves with the learning rate at fixed weight decay would count against it.
- Sequence models keep the fraction of their layers that use full attention at 0, at 1, or near a small-denominator level (1/2, 1/3, 1/4, 1/6 or 1/8), scored by the distance to the nearest level against a null; the level set is a scoring convention, not a consequence of concave pricing (Section 3.5). This is the list's weakest test.
- In looped blocks without input re-injection, the fraction of a content mode that survives falls tenfold after about $2.3/(p(1-\mu_C))$ visits, with the mixing share p and the content eigenvalue μ_C measured rather than fitted.
- Mixers that only transport, mixing each group's tokens by one shared orthogonal matrix and so preserving the group's channel-by-channel inner products (its channel Gram matrix), underperform on retrieval-and-write tasks at any depth. In a provisional test, not parameter-matched, adding a writing channel restored classification but not dictionary retrieval.

Each can be checked with public models or small controlled runs.

6.3 Limits

Scale. The controlled evidence reaches 125 million parameters and is mostly below 25 million. Evidence at production scale is measurement on public checkpoints, and further decisive tests in trained language models remain pending (Appendix B.4).

The price model. Calling the protections "cheap" (Section 3.3) is a judgment under a semi-formal price model, not a computation. We did not measure the alternatives' prices on a common budget of hardware and training, and the prices themselves co-evolved with the transformer as hardware, kernels and recipes adapted to it, so the sweet spot is a reading under that model, not a measured optimum.

Retrodiction. Reading past design changes through the conditions (Section 3.6) is an interpretation; only predictions made in advance (Section 6.2) can test it.

Decoders and the MLP. The vocabulary of stationary laws, reversal and circulation (Section 4.1) needs a positive stationary law, which causal softmax attention lacks. Causal softmax attention takes its direction from the mask. For decoders we use only statements proved for them, such as the absorbing first token and rollout as survival (Section 4.2) and the two-sided law (Section 3.7). Separately, there is no theory here of what the MLP should compute, beyond that it writes (Section 3.2) and that it is the main resource that compensation draws on (Section 3.4).

A conditional inventory. The conditions hold within their declared classes (Section 3.2) and say nothing outside them. Consider how content survives depth. For normalized nonnegative filters of a fixed mode $0 < \mu < 1$, content survives depth in only one way, by keeping short-path mass. The two other ways, signed coefficients and invertible transport (Section 4.5), work by leaving that class. Without the fixed-mode premise, even this list of three is incomplete: an operator that moves toward the identity as depth grows keeps content with none of the three, because the operator itself changes with depth (M§14.2).

Idealized settings. Many results are proved for one operator applied repeatedly; Section 5 says which, and what has been measured in trained networks.

6.4 Components and roots

Most protections in use were found to be needed when a component broke: pre-normalization when deep post-norm models would not train (Wang et al., 2019), normalized queries and keys when attention logits grew at scale (Dehghani et al., 2023), sink tokens when evicting the first tokens broke streaming generation (Xiao et al., 2024), registers when vision transformers showed high-norm artifact tokens (Darcet et al., 2024), log-N scaling and temperature when models failed beyond their training length (Su, 2021; Chiang & Cholak, 2022; Peng et al., 2024), and hybrids when pure recurrences struggled to recall (Arora et al., 2024a;b). Component work built the transformer, but it does not by itself say why most modifications barely help, which parts are essential, or where to look next. Asking what each component *protects* does so, and in this collection that question kept proving fruitful. Asking why sinks appear led to one common form, a gate times a softmax, for every zero-valued null patch and headwise output gate, and to the two-sided law. Asking why the residual cannot be removed led to path laws and to the anchor exception. The missing anchor is a deficit that no capacity of bounded gain (bounded sensitivity) can make up for under strict mixing, the only one so proved; in the small-model ablations nothing else in the block made up for it, and a test that adds capacity is pending. Asking whether the value map matters led to compensation. Asking what bounds attention's loop gain led to a certificate, and to a test showing that divergence flags can mistake the anchor's relaxation for feedback. Asking whether an asymmetric attention matrix is directional led to circulation and the parity law. These two kinds of work, fixing components and asking what they protect, are complementary, and the second is slow. Its negative results count as evidence: the results for explicit protection knobs at small scale (Section 3.4) support the reading that a tuned standard block already supplies its protections, and compensation was found because necessity predictions kept failing in trained networks. The transformer was built by fixing what broke; asking what each fix protected, and keeping what failed along the way, is how we began to learn what holds it in place.

This project's ambition has outlasted my endurance. An account this broad is surely wrong in places, partly because it is this broad, so I would ask readers to poke and prod at it; the predictions stated in advance (Section 6.2) and the tests still pending (Appendix B.4) are the quickest ways to find where. Part of why the transformer abides is fast matrix multiplication, engineering and history; part, I have argued, is that it pays cheaply for what attention needs. That explains its past, not its future. Transformers will keep changing where room remains, and an architecture that needs none of this, or buys it more cheaply, need not look anything like one.

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

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, 2020.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4895–4901, 2023.

Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Ré. Zoology: Measuring and improving recall in efficient language models. In *International Conference on Learning Representations*, 2024a.

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, 2024b.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*, 2015.

Stefan Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3:133–181, 1922. doi: 10.4064/fm-3-1-133-181.

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. Curran Associates, Inc., 2022.

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Petar Veličković, Razvan Pascanu, and Michael Bronstein. Why do LLMs attend to the first token? In *Conference on Language Modeling*, 2025.

Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Quantizable transformers: Removing outliers by helping attention heads do nothing. In *Advances in Neural Information Processing Systems*, 2023.

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.

Neil D. B. Bruce. Beyond transformers: Sums, maxima and the price of order. Zenodo, released with this paper, 2026e. URL https://doi.org/10.5281/zenodo.23542133.

Neil D. B. Bruce. Lost in ranking: When attribution scores hide class contrast. Manuscript, 2026f.

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.

Aakanksha Chowdhery et al. PaLM: Scaling language modeling with Pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

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

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10041–10071. PMLR, 2024.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16344–16359. Curran Associates, Inc., 2022.

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

DeepSeek-AI. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model. arXiv preprint arXiv:2405.04434, 2024.

Mostafa Dehghani et al. Scaling vision Transformers to 22 billion parameters. In *International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7480–7512. PMLR, 2023.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning (ICML)*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Nelson Elhage et al. A mathematical framework for Transformer circuits. Transformer Circuits Thread, https://transformer-circuits.pub/2021/framework/index.html, December 2021.

Alexander M. Fichtl, Jeremias Bohn, Josefin Kelber, Edoardo Mosca, and Georg Groh. The end of Transformers? On challenging attention and the rise of sub-quadratic architectures. arXiv preprint arXiv:2510.05364, 2025. Revised version published as "Challenging Quadratic Attention – A Holistic View On the Rise of Alternative Language Model Architectures", Proceedings of The Big Picture v2: Crafting a Research Narrative, pages 60–81, Association for Computational Linguistics, 2026.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach. In *Advances in Neural Information Processing Systems*, volume 38, pp. 41340–41391. Curran Associates, Inc., 2025.

Gemma Team. Gemma 2: Improving open language models at a practical size. arXiv preprint arXiv:2408.00118, 2024.

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.

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.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. In *International Conference on Learning Representations*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Alex Henry, Prudhvi Raj Dachapally, Shubham Shantaram Pawar, and Yuxuan Chen. Query-key normalization for Transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4246–4253. Association for Computational Linguistics, 2020.

Jordan Hoffmann et al. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30016–30030. Curran Associates, Inc., 2022.

Sara Hooker. The hardware lottery. *Communications of the ACM*, 64(12):58–65, 2021.

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.

Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming*, 127:203–244, 2011.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361, 2020.

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 P. Kelly. *Reversibility and Stochastic Networks*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, Chichester, 1979. URL https://www.statslab.cam.ac.uk/~frank/BOOKS/kelly_book.html.

Hyunjik Kim, George Papamakarios, and Andriy Mnih. The Lipschitz constant of self-attention. In *International Conference on Machine Learning (ICML)*, 2021.

Kimi Team. Kimi Linear: An expressive, efficient attention architecture. arXiv preprint arXiv:2510.26692, 2025. Model card: https://huggingface.co/moonshotai/Kimi-Linear-48B-A3B-Instruct.

A. Kolmogoroff. Zur Theorie der Markoffschen Ketten. *Mathematische Annalen*, 112(1):155–160, 1936. doi: 10.1007/BF01565412. URL https://doi.org/10.1007/BF01565412.

Maxime Labonne. Qwen3.5: Nobody agrees on attention anymore. Hugging Face community article, https://huggingface.co/blog/mlabonne/qwen35, February 2026.

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

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, 2025.

William Merrill and Ashish Sabharwal. The expressive power of Transformers with chain of thought. In *International Conference on Learning Representations*, 2024.

Evan Miller. Attention is off by one. Blog post, https://www.evanmiller.org/attention-is-off-by-one.html, July 2023.

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

Trishit Mondal and Ameya D. Jagtap. In transformer we trust? A perspective on transformer architecture failure modes. arXiv preprint arXiv:2602.14318, 2026.

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

Sharan Narang et al. Do Transformer modifications transfer across implementations and applications? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5758–5773. Association for Computational Linguistics, 2021.

Catherine Olsson et al. In-context learning and induction heads. Transformer Circuits Thread, https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html, March 2022. Also arXiv:2209.11895.

Kenta Oono and Taiji Suzuki. Graph neural networks exponentially lose expressive power for node classification. In *International Conference on Learning Representations (ICLR)*, 2020.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. Model card, https://openai.com/index/gpt-oss-model-card/, August 2025. Released 5 August 2025; also arXiv:2508.10925.

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.

Zihan Qiu et al. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. In *Advances in Neural Information Processing Systems*, volume 38, pp. 100092–100118. Curran Associates, Inc., 2025.

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.

R. Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.

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.

Noam Shazeer. Fast transformer decoding: One write-head is all you need. *arXiv preprint arXiv:1911.02150*, 2019.

Jianlin Su. Viewing the scale operation of attention from the perspective of entropy invariance. Scientific Spaces (blog), https://kexue.fm/archives/8823, December 2021. In Chinese; English translation of the original title.

Mingjie Sun, Xinlei Chen, J. Zico Kolter, and Zhuang Liu. Massive activations in large language models. In *Conference on Language Modeling*, 2024.

Rich Sutton. The bitter lesson. Blog post, http://www.incompleteideas.net/IncIdeas/BitterLesson.html, March 2019.

Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Tran, Dani Yogatama, and Donald Metzler. Scaling laws vs model architectures: How does inductive bias influence scaling? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 12342–12364. Association for Computational Linguistics, 2023.

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, 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.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep Transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1810–1822. Association for Computational Linguistics, 2019.

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.

Zhenda Xie et al. mHC: Manifold-constrained hyper-connections. arXiv preprint arXiv:2512.24880, 2025.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the Transformer architecture. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10524–10533. PMLR, 2020.

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., 2024.

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

Tianzhu Ye, Li Dong, Yuqing Xia, Yutao Sun, Yi Zhu, Gao Huang, and Furu Wei. Differential Transformer. 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.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M. Susskind. Stabilizing Transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 40770–40803. PMLR, 2023.

A The monograph and how to read it

This paper rests on a monograph (Bruce, 2026a) that develops each of this paper's results with its evidence and its scope, except two remarks marked as unreleased work and three results whose full record it defers to a later version (Section 1), and maps the collection of papers from which those results come (M§R.1). Three of those papers are released with this one: on the length laws of softmax for retrieval and abstention (Bruce, 2026c), on the transformer as a constrained equilibrium of protections (Bruce, 2026d), and on what machines beyond the transformer pay for the same conditions (Bruce, 2026e). The collection's other results are not yet released as papers, and this paper cites them through the monograph's sections.

A.1 Reading paths

The monograph's reading guide maps the argument of this paper to the chapters that develop it. The paths below cut across that map by topic. Each is a few chapters, read in order.

Long context. Chapter 8 shows why weights that sum to one leave a head no way to attend less, and that every zero-valued null patch (such as a zero-value sink or register) or headwise output gate factors as a gate times a softmax. Chapter 9 adds the routing budget: the score lead the right key needs over the other keys, which grows with length. Chapter 17 puts abstention and retrieval together at long context, from a controlled pilot to a production model and trained models up to 125M parameters, with the 350M-parameter models still to run. Chapter 19 prices the same conditions in other machines' currencies. M§6.4 shows how small leaks of attention across the boundary between blocks of tokens, such as the documents packed into one context, can grow with path length, while exact block separation keeps them at zero.

Recurrent and hybrid models. Chapter 14 sets the transformer's constraints in a lattice of designs and shows how relaxing one obliges a protection. Chapter 18 follows recurrent and streaming mixers through that lattice: a common contraction factor keeps state bounded when keys and values are bounded; a linear memory's per-head capacity limits exact recall of arbitrary associations, however far back an item lies; and hybrids keep a reserve of attention layers. Chapter 19 compares the currencies in which machines pay, and M§20.4 registers the tests still open, among them a sweep of the fraction of attention layers in a hybrid.

Depth, residuals and normalization. Chapter 1 reads one averaging operator as a walk. Chapter 2 shows that depth is a distribution: for one fixed operator, what survives many layers depends on how often the signal takes a short route; the mixing budget sets how fast the rest is lost; and at initialization, normalization placement is a choice of that budget. Chapter 5 gives the anchor and its removal signatures, and shows that under strict mixing its loss is a deficit that no capacity of bounded gain (bounded sensitivity) can make up for, the only one so proved. Chapter 15 shows that a small trained network can evade a necessity theorem when another resource has room and its retained state still separates the task's outputs. M§§16.6–16.7 assemble the smallest block that meets the conditions, with the evidence recorded for it and the tests still to run.

Stability, fixed points and looped models. Chapter 7 gives the condition that holds feedback below the anchor of Chapter 5: it proves the anchored-contraction certificate, computes softmax's feedback constant and reads weight decay as an anchor. Chapter 12 shows that a converged average can still amplify changes in its inputs and overturn the predictions read from it. Chapter 13 shows that a computation which settles to machine precision certifies neither its response to the input, nor the uniqueness of its endpoint, nor that the endpoint attracts, nor that a loop built on it is stable; the chapter also gathers a battery of checks for fixed-point computations. These results are proved and shown on constructions and small systems; trained looped and implicit models are still to be tested.

Direction. Chapter 3 splits every head with a positive stationary law into preference, exchange and circulation, locates direction exactly, and shows why default statistics of a single operator cannot see it. Chapter 10 takes direction up as a condition, since a head without circulation cannot shift tokens around a cycle and routes reliably along an open sequence only by piling its stationary mass toward one end. The chapter measures circulation in pretrained models, shows that the learned orientation is consequential, and

gives the price of order: how much of its score range a soft router must spend to tell a later write from an earlier one.

Evidence and method. The reading guide's account of the evidence (M§R.2) comes first. M§15.4 reports a preregistered study in which a protection made into an explicit knob did not beat the block's own mechanism. M§4.4 asks when propagation helps frozen features, since a strong signal need not be useful. Chapter 20 gives the predictions stated in advance and the register of open tests, and the monograph's Appendix B gives the evidence ledger; their short forms are in Appendices B and C.

A.2 Where to find it

This paper is on arXiv. With the monograph and the three papers named above, it forms one collection on Zenodo. The monograph's text is licensed under the Creative Commons Attribution 4.0 International licence. To cite a single result, cite the monograph and its section, as this paper does with M§; to cite the argument as a whole, cite this paper. The records behind the numbers, the code that produced them and the formal proofs follow in a later version of the collection: each paper's package is archived as its own Zenodo record and cross-linked to the monograph. The monograph's first version, 0.1, has the DOI `https://doi.org/10.5281/zenodo.23541723`; its Zenodo record links to any later version.

B Evidence standards, and what is still pending

B.1 Labels and tiers

Each of the collection's claims carries one of five evidence labels, which say what kind of support it has. A claim is *proved* when it is a theorem with a text proof within a declared class, in most cases checked numerically and, for some, formalized in Lean; *measured* when it is measured on records that exist now, in controlled runs or on public checkpoints; *partial* when there is real evidence and a decisive proof or run is still owed; and *pending* when it is stated or predicted and awaits evidence. An *interpretation*, such as the sweet spot under the price model (Section 3.3) or the retrodiction (Section 3.6), is a reading of the evidence under a stated model and is never counted as evidence. The usual answers of Section 2 carry a sixth label, *conventional*: an explanation from the current literature that the collection does not test. Each claim also carries a transfer label, which says what carries over to trained networks (Section 5).

A measurement carries a tier, which says what it was made on. **SYN** is a law checked on synthetic constructions; **SYN-R** is a law checked on real operators or spectra along an artificial perturbation; and **REAL** is a measurement of trained models or real data, including models trained on synthetic tasks. A fourth tier, **ID**, is a machine-precision check of an identity: it certifies algebra and code, not a claim about models.

Claims about practice carry two further labels that distinguish after-the-fact readings from tests stated in advance. A *retrodiction* reads an adopted change after the fact (Section 3.6) and is never evidence for the account. A *prediction* is stated before any test of it (Section 6.2); only tests made after it is stated can count as evidence for it (Table 5).

B.2 Preregistration and dispositions

The collection includes preregistered experiments and prespecified analyses, registered in the repository before their runs rather than in an external public registry. Each experiment's registration fixes its criteria and, where it tests a prediction, its falsifier before it runs; amendments are recorded beside the registration, and a verdict, once written, is never overwritten. Where a later audit corrected a record or an assessment, the frozen verdict is retained as provenance, with the correction stated beside it. A statement is established when it is proved, with its hypotheses stated, or when an experiment preregistered with a falsifier returned the registered outcome on its frozen criteria and its validity checks held at the tier given. Verdicts use the program's vocabulary: *pass*, *fail*, *mixed*, *inconclusive*, *void*, *withdrawn* and *rescoped*, with *data-absent* for a leg (a planned part of an experiment) whose inputs never arrived. A void or data-absent leg is never read as

a negative. For an inconclusive result, the account states what the measurements show and what remains unresolved, including any trend they support. Tolerances and pass-fail checks are likewise fixed before a run; corrections to defective checks are recorded as amendments, with the original criteria and verdicts retained. Short sanity-check smoke runs are excluded, and where a saved summary assessment is stale, the quoted numbers are recomputed from the per-cell records (Bruce, 2026a;d, Appendix B).

B.3 Proof checking

Most identities and laws stated in this paper are checked numerically, and several are formalized in Lean. The evidence behind each claim, including selected check counts, is recorded in the package's claims export (`data/claims`), released with the package in a later version of the collection. A numerical check confirms a statement on the cases it checks, not in general; a text proof, and a Lean formalization where one exists, establish the statement within the class it declares, and that class is stated with each result.

B.4 What is still pending

Table 5 lists the decisive tests, with what each would change. Two have returned (Section 3.7): the prespecified test of a production model's sinks, which was undecided by its rule (descriptively, its slopes lean toward compensation through position), and the trained language-model test of abstention rules at 125M, where the $\log N$ tempering prediction failed on top of YaRN, while every rule that gives attention a null removed the first-token sink and most massive activations, as that test's registration predicted for language models and as earlier reports found; the 350M leg and the other four tests await their main results. The claims ledger (Appendix C) lists them with their transfer notes and the sections of the monograph that develop them.

Table 5: Decisive tests: what each measures, its status, and what each outcome would change in this paper. Generated from the claims ledger.

Test	What it measures	Status	What each outcome would change	Chapter
Do a production model's learned sinks stop abstaining with length?	Measures whether gpt-oss-20b's sink log-odds fall with context length in dense layers and stay flat in banded ones beyond their window.	*measured*. Undecided by its prespecified rule (its books control failed, and its slopes matched neither branch), though descriptively its slopes lean toward compensation through position.	Confirm: the account of abstention at long context (Section 3.7) gains a production-model confirmation, stated before the run. Compensated: background logits fall with length; the law holds but trained models adapt. Inconclusive: the controlled pilot and the trained harness carry the evidence.	17, 20
Which abstention rules keep both sides in trained language models?	Trains 125M language models with each abstention rule, including the max-statistic head, and measures abstention and retrieval out to long context. The learned sink and the gate with a learned sink are also tempered at evaluation.	*measured*. Returned at 125M, with the 350M leg not yet run: every rule with a null removed the first-token sink; the lead prediction failed, as tempering on top of YaRN, which already carries a log-length temperature, raised long-context degradation and did not hold abstention; and the max-statistic head was the only rule to keep both weights within half a nat on the seed means with YaRN, and failed natively.	Decides whether tempering is stated as a measured fix at scale, and whether the max-statistic head escapes the log-N length schedule (Sections 3.7 and 4.4).	17, 20
What fraction of attention layers does a qualified recurrent hybrid need?	Sweeps the fraction of attention layers in a recurrent hybrid with certified contracting writes at 125M and 350M against qualified baselines that pass the recall check.	*pending*	Decides whether bounded memory with an attention reserve is a resource-axis win (such as memory or speed) at quality parity (Section 3.7).	18, 20
Does the minimal block's accuracy margin survive tuned baselines?	Replicates the 22M-parameter comparison with a clean implementation against DeiT-recipe baselines.	*pending*. Reruns of historical cells matched the recorded accuracies exactly (0 pp); the tuned replication has not run.	Survives: the blueprint (Section 3.6) gains an at-scale data point. Shrinks to parity: stated as parity; no win over tuned transformers is claimed. Loses: the 22M result is reported as not replicated, and the blueprint rests on the small-scale record.	16, 20
Does compensation grow with scale?	Measures the capacity a tied value map costs against MLP ratio, width and depth in vision transformers on ImageNet-100 and a 125M language model.	*pending*	Tests the proposed mechanism for why alternatives' small-scale improvements fade (Section 3.4).	15, 20
Can any capacity replace a removed anchor?	Removes the identity path and sweeps MLP width, heads and depth against arms that re-supply an anchor another way.	*pending*	Tests the anchor exception in trained networks (Section 3.4).	5, 20

C Claims ledger

Table 6 lists every claim in this paper and the decisive tests, with their evidence and transfer labels, the sections of the monograph that develop each, and, where they exist, the program's records behind each. The table is generated from the public fields of the claims ledger, which also generates Tables 1 to 5. The claims export (`data/claims`), released with the package in a later version of the collection, holds each claim entry in full, with its plain and technical statements, its evidence and its caveats where supplied.

Table 6: Claims ledger: every claim in this paper and the decisive tests, grouped by the section that makes each. The columns give each claim's evidence label (Appendix B), its transfer label and scope note (Section 5), the sections of the monograph that develop it (Bruce, 2026a), and the program's records behind it with their evidence tier (`SYN`, `SYN-R` or `REAL`, Appendix B); a tier without an identifier marks evidence that has no separate program record. The usual answers carry the transfer label literature: explanations from the literature, not tested by the collection, with what the collection adds to each in Table 1. Decisive tests that have not returned carry the label pending: each will be labelled when it returns. A dagger marks an exact identity: it holds by construction for every operator of its class, and its value is what it makes measurable. Generated from the claims export.

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
The usual answers (Section 2)				
It fit the hardware: large dense matrix multiplications	conventional	literature	§14.4	—
Scaling laws made capability predictable and purchasable	conventional	literature	§21.4	—
The ecosystem co-evolved around it	conventional	literature	§14.2	—
Proving an alternative at scale costs too much	conventional	literature	§§15.6, 18.5	—
History: the right objective, data and hardware in one place	conventional	literature	§16.5	—
We measure what transformers are good at	conventional	literature	§3.5	—
In-context learning turned one model into a platform	conventional	literature	§21.4	—
Weak assumptions plus abundant data win	conventional	literature	§1.1	—
Training is parallel, and every position gives a signal	conventional	literature	§18.4	—
Exact, content-addressable memory makes recall possible	conventional	literature	§§9.6, 18.3	—
The residual stream is a bus; blocks communicate, then compute	conventional	literature	§§5.1, 11.7	—
It trains: conditioning is half the design	conventional	literature	§§2.6, 5.3, 7.5	—
Data-dependent routing built from matrix multiplications	conventional	literature	§§1.3, 14.4	—
Softmax can sharpen into a pointer	conventional	literature	§§9.1, 9.4	—
The architecture is not the algorithm	conventional	literature	§15.1	—
It absorbs its challengers as components	conventional	literature	§§16.5, 19.6	—
Challengers converge towards attention	conventional	literature	§§18.1, 19.6	—
Weaknesses get worked around above the architecture	conventional	literature	§§18.1, 21.4	—

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
The design space around it is flat	conventional	literature	§15.4	—
Scale sorts architectures, so far in the transformer's favour	conventional	literature	§15.6	—
The transformer endures as a template for learned computation	conventional	literature	§14.5	—
A sweet spot of conditions (Section 3)				
A mixing layer is an averaging operator rebuilt from its input	proved	exact: Every softmax head, at every layer and input, is a row-stochastic kernel $P(X)$ whose outputs are convex combinations of its values. The path-lottery form $\mathbb{E}[P^K]$ describes powers of one operator; a trained stack is an ordered product of different $P(X)$, with MLPs between.	§§1.1, 1.2, 1.3, 1.4	—
The same laws govern graphs, denoisers and attention	proved	idealized: The collapse law is stated for powers of one fixed operator, which linear propagation over a fixed graph satisfies exactly; the certificate is stated for one builder $P(\cdot)$ applied repeatedly, as in iterative denoisers, deep equilibrium models and test-time mixing. A transformer's forward pass applies a different $P(X)$ at each layer.	§§4.1, 4.2, 4.4	E3.2, N30 (SYN-R, REAL)
Deep averaging forgets unless some paths stay short	proved	idealized: The if-and-only-if is proved for path laws over powers of one fixed operator. Bounds for realized products of distinct layers and a Dobrushin bound for sink-like attention carry its direction, that mixing without short paths forgets, to stacks whose operators change from layer to layer.	§§2.2, 2.3, 2.6, 5.1, 5.2, 5.3, 5.4, 5.6	N25, E04, N37, SX43 (REAL)
Routing and content need separate maps	proved	exact under premises: Exact on the identified block model where necessity is stated; whether a trained network sits in that model must be checked, and with spare capacity trained networks evade it (compensation).	§§6.1, 6.2, 6.3, 6.6	N12b, N11, E2.2 (REAL)
Self-reinforcing weights need their feedback held below the anchor	proved	exact under premises: The per-layer bound $\kappa_f \le 2sR^2$ (at logit scale s and state radius R) holds for every softmax layer with tied dot-product scores (projected scores multiply it by the projections' norms). Contraction and a single attracting state are statements about one builder applied repeatedly, exact once the contraction margin $m > 0$ is computed on it: weight-tied loops and deep equilibrium models are in that class, a stack of distinct layers applied once each is not.	§§7.1, 7.2, 7.5, 7.6, 12.2	N31, N33 (SYN)
Attention cannot abstain unless it is given a way out	proved	exact: Unit row sums, the kernel of the softmax Jacobian (a uniform logit shift changes no weight) and the gate-times-softmax factorization hold for every head, at every layer and input.	§§2.7, 8.1, 8.2, 8.4, 8.5, 8.6	N19, R7c, RIII1 (REAL)
One key among N needs a score margin growing like $\log N$	proved	exact under premises: Exact for any softmax row whose logits lie in $[-\tau, \tau]$: the bound is worst case, and the logit bound τ is read off the network.	§§9.1, 9.2, 9.3, 9.6, 17.2, 17.4	R5d, V1, SX40, SX02, SX113 (REAL)

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
A head that cannot circulate cannot shift tokens around a cycle	proved	exact: An identity for every circulation-free head with a positive stationary law, on cycles whose successor and predecessor probabilities are positive, at any layer, stated per head; compositions of reversible heads can acquire direction, and causal heads take theirs from the mask.	§§10.1, 10.2, 10.5, 10.6	N13 (REAL)
Moving information around is not enough; something must write	proved	exact: Holds for every row-stochastic mixer, at every layer and input; the Gram invariant holds for every shared orthogonal mixing of a token group.	§§11.1, 11.4	ATS07, ATS12 (REAL)
We read the standard block as paying for every condition at once	interpretation	interpretation: A reading under a semi-formal price model; there is nothing to transfer.	§§14.1, 14.5	AF10, AF02, BD30 (REAL)
Hardware sets the prices; the conditions are what is priced	interpretation	interpretation: A reading that joins the hardware account to the conditions; there is nothing to transfer.	§§14.4, 19.6	—
Every machine pays to name one of N, in some currency	proved	exact under premises: Each part is exact within its class: Fano's bound and the order bound for any decoder; the logit-range (expression) bound for any soft majority; the memory floor for one-pass states with arbitrary later queries over unstructured items. The key-channel law assumes random keys on a sphere.	§§9.5, 19.2, 19.3	SX29, SX57, SX58, SX59 (SYN)
Relaxing a constraint buys a capability and obliges a re-supply	interpretation	interpretation: A reading of the lattice; the individual re-supply obligations are proved in their chapters where stated.	§§14.2, 14.3	—
Trained networks buy back what a component lacks	measured	measured: A mechanism measured in trained networks: necessity theorems bind exactly on their constructions, and trained networks evade them when a resource is spare.	§§6.6, 15.1, 15.2, 15.3	N24b3, N12b, N12, BD30, E04b, SX48 (REAL)
No capacity of bounded gain buys back short paths	proved	exact under premises: Holds for any stack whose layer maps restrict to a common discriminative subspace with norms at most one and whose contraction product tends to zero, with the recovery gain uniformly bounded (a Dobrushin version covers sink-like attention with shared feature maps); these premises are checked layer by layer.	§5.5	N25 (REAL)
Making a protection explicit does not beat a block that has it	measured	measured: Measured on twelve preregistered interventions in standard trained models.	§15.4	AF01, AF02, AF04, AF08, AF10 (REAL)
Deficits may cost less as models grow	pending	pending: A hypothesis about trained networks at scale; its test is pending.	§15.6	—
Under concave prices, the best design is a corner	proved	exact under premises: A theorem about allocations under concave prices; its premise, concave pricing of protections at the level of architectures, is assumed, not measured.	§§16.1, 16.2, 16.3, 16.4	N35, E5.4 (SYN, REAL)
Read backward, adopted changes re-supply the same protections	interpretation	interpretation: A reading of adopted changes; there is nothing to transfer.	§16.5	—
Read forward, the conditions rebuild a transformer	interpretation	interpretation: A reading of an assembled constraint set.	§16.6	N24, N24b2 (REAL)
A 22M-parameter accuracy margin has not yet been replicated	partial	measured: Recorded at 22M parameters with 5 paired seeds against untuned baselines; not evidence of dominance until the tuned replication returns.	§16.7	E2.1, SX49 (REAL)

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
No fixed rule keeps both abstention and retrieval at long context	proved	exact under premises: The impossibility bound is exact for any softmax row with content logits in $[-\tau, \tau]$ and a null shared by absent and present queries; tempering is guaranteed to keep both sides at every length only under strictly separated margins, which are checked on the head, and over a finite range it can keep both without them, as in the pilot.	§§17.1, 17.2, 17.6	SX40, SX02, SX113, SX114 (REAL)
Bounded memory buys length robustness and pays in recall	proved	exact under premises: Exact for linear memories (through the rank of the keys) and for delta-rule writes with bounded values and decay under the stated step-size condition, for any sequence of keys bounded in norm.	§§18.1, 18.2, 18.4, 18.6	ATS04, N36, SX47, ATS11 (REAL)
Decisions on maxima need no length schedule	partial	idealized: Shown for constructed keys and backgrounds (independent or exchangeable scores).	§§17.4, 17.5	SX02 (SYN, REAL)
What makes a good writer is still open	pending	pending: Open: there is no theory yet to transfer.	§§11.7, 19.6, 21.3	—
What any successor must still supply	interpretation	interpretation: A checklist read from the stated classes; there is nothing to transfer.	§§19.1, 19.6	—
Most controlled evidence is small	interpretation	interpretation: A statement of scope.	§21.2	—
'Cheap' rests on a price model, not measured prices	interpretation	interpretation: A statement of scope.	§§14.5, 21.2	—
Explaining the past is not evidence	interpretation	interpretation: A statement of scope.	§§16.5, 21.2	—
What the question uncovered (Section 4)				
Every unmasked attention head splits into preference, exchange and circulation†	proved	exact: The split is an identity for every Markov kernel with a positive stationary law; for every unmasked softmax head with finite scores, its score asymmetry also splits into a sink and a cycle, at every layer and input. Causal heads have no positive stationary law.	§§3.1, 3.2, 3.3, 3.4, 10.3	—
Trained attention's direction is specific, and it matters	measured	measured: Measured in trained encoders.	§§10.3, 10.4	RCE02AAB, RCE05 (REAL)
Standard diagnostics of a single operator cannot see which way attention points	proved	exact under premises: Exact, for stationary-weighted quadratic readouts, for one amplified operator and for palindromic words of layers that share a positive stationary law. A stack of distinct layers is not a palindrome, and its readouts generally move at first order (measured median log-log slope 1.000 in the circulation strength, against 2.000, for 48 matched pairs of non-palindromic and palindromic five-layer words of balanced BERT layers from eight inputs).	§§3.5, 3.6, 10.2	RC3 (REAL)
Depth is a distribution: short paths decide what survives	proved	idealized: Proved for path laws over powers of one fixed operator; exact for propagation on a fixed graph, not for the full nonlinear computation of a trained network, the scope its source paper states.	§§2.1, 2.2, 2.3, 2.5, 2.8	—
The expected number of mixing layers sets how fast depth forgets	proved	idealized: Stated for expected energies of residual stacks with random feature maps and mean-field normalization.	§§2.4, 2.7	(SYN)

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
Deeper trained language models send their surplus attention to the sink	partial	measured: Measured on rollout path laws, a linearization of attention flow built from each model's own attention and residual norms.	§§2.5, 2.7	SX60 (REAL)
Attention rollout has an exact probabilistic meaning†	proved	exact: Exact for ordered products of causal stochastic kernels (the first token absorbing) with per-token residual shares, whatever the layers: layer-varying by construction. It describes the rollout linearization, not the full block.	§8.3	(REAL)
A sharp, machine-checked stability test for adaptive averaging	proved	exact under premises: Exact for any anchored adaptive iteration once the contraction margin $m > 0$ is computed on it: weight-tied loops, deep equilibrium models, iterative denoisers and test-time mixing. A stack of distinct layers applied once each is not in the class.	§7.1	—
A settled average can amplify changes and destroy accuracy	measured	measured: Measured in iterated systems that apply one builder repeatedly (softmax kernels, a real denoiser, test-time mixing over 32-image contexts); the maxima were selected from grids.	§§12.1, 12.2, 12.3, 12.4	OUB04, OUB05 (REAL)
Each prediction's class margin splits into its own and its companions'†	proved	exact: An identity for any mixing followed by an affine readout, at every iteration and at the fixed point.	§§12.4, 12.5, 12.6, 12.7, 12.8	OUB05 (REAL)
Converging is not the same as being right or stable	partial	exact: A negative result that holds for any solver: settling certifies none of response, uniqueness, attraction or outer-loop stability; shown by explicit constructions.	§§13.1, 13.2, 13.3, 13.4, 13.5, 13.6, 13.7, 13.8, 13.9, 13.10	OUB01, OUB12, OUB13 (SYN)
One stability threshold links logit scale to weight decay	proved	exact under premises: The loop-gain bound holds for every softmax layer with tied dot-product scores (projected scores multiply it by the projections' norms); the cap assumes that decoupled weight decay acts as the anchor (step share $\alpha = \eta$ and anchor share $\beta = \eta\lambda$ at learning rate η and weight decay λ), a modelling assumption.	§§7.3, 7.4, 7.7	FB9, SX95 (REAL)
Softmax's $\log N$ law: abstention and retrieval cannot both be fixed	proved	exact under premises: Exact for any softmax row with content logits in $[-\tau, \tau]$ and a null shared by absent and present queries, at any layer; tempering needs strictly separated margins, checked on the head.	§§9.1, 9.2, 17.1, 17.2, 17.4	SX40, SX112 (REAL)
Sums need a length schedule; maxima do not	partial	idealized: The argmax bound needs a uniformly placed target and a joint likelihood that depends on the target only through a nondecreasing factor of its own score; independent backgrounds with a monotone likelihood ratio satisfy this, exchangeable ones need not. The onset law assumes independent keys; the tests are constructed.	§§8.4, 9.3, 9.4, 17.5	(SYN)
Soft routers pay for order out of their score range	proved	exact: A combinatorial bound for any soft router whose positional logit term has range B; its cost to the score margin, when every position must stay retrievable, is a mean-field statement.	§10.7	—
Small attention leaks can grow with depth; exact block zeros stop them	proved	idealized: The growth bound is stated for path laws of one operator with a per-row crossing probability; exact block zeros stay zero for any stack whose layers share the blocks.	§§6.4, 6.5	(SYN)

Claim	Evidence	In trained networks (transfer label and note)	Monograph	Records
A fixed memory's recall is limited by capacity, however far back	proved	exact under premises: Exact for linear read-back from a d_k-dimensional memory (through the rank of the keys) and, for the state bound, for writes that contract by a common factor.	§18.3	ATS04 (REAL)
Signed attention is already inside your transformer	partial	exact: A representation identity: two stochastic heads give any signed kernel with constant row sums.	§15.5	BD30, BD32, BD33 (REAL)
Invertible blocks keep capability; the tested orthogonal transport loses it	partial	measured: Measured on trained mixers; the invariant behind it is exact (shared orthogonal transport preserves each token group's channel Gram matrix), but it does not imply the retrieval failure.	§§11.2, 11.3, 11.4, 11.5	ATS07, ATS12, ATS13 (REAL)
Production vision transformers grow sinks that are not empty	measured	measured: Observational measurement in pretrained models.	§8.5	SX44 (REAL)
A strong signal is not the same as a useful one	measured	measured: Measured with frozen features and label propagation on neighbour graphs.	§4.4	E01e, E01d (REAL)
What this changes (Section 6)				
Predictions stated before the evidence	pending	pending: Predictions about trained networks, stated before their tests; the tests already run are reported with their outcomes.	§20.1	—
Decoders and the MLP are only partly covered	interpretation	interpretation: A statement of scope.	§§10.8, 11.7, 21.2	—
The account says nothing outside its class or its prices	interpretation	interpretation: A statement of scope.	§21.2	—
The decisive tests (Table 5)				
Do a production model's learned sinks stop abstaining with length?	measured	measured: Prespecified on gpt-oss-20b: undecided by its prespecified rule; descriptively, its slopes lean toward compensation through position.	§§17.3, 20.2	SX01 (REAL)
Which abstention rules keep both sides in trained language models?	measured	measured: Prespecified 125M language-model test: its lead prediction (that log N tempering on top of YaRN holds abstention and lowers degradation) failed. Every rule that gives attention a null removed the first-token sink and cut massive activations, all but sigmoid lowering perplexity at the training length; positional interpolation carried most of YaRN's gain past the training length, and log-N tempering helped on interpolated positions. In its vision subset (ViT-S/16 on ImageNet-100), tempering raised accuracy at untrained token counts without restoring the trained attention. The abstention results describe attention weights, not behaviour.	§§17.4, 20.4	SX02, SX114 (REAL)
What fraction of attention layers does a qualified recurrent hybrid need?	pending	pending	§§18.4, 20.4	—
Does the minimal block's accuracy margin survive tuned baselines?	pending	pending	§§16.7, 20.4	—
Does compensation grow with scale?	pending	pending	§§15.6, 20.4	—
Can any capacity replace a removed anchor?	pending	pending	§§5.5, 20.4	—