The Geometry of Forgetting: Toward a Law of Information Decay in Self Modifying Systems

Jace Hall[1]

[1]**Hall iNtelligence, LLC**

ABSTRACT

Self modifying systems suffer not only from catastrophic forgetting but from a deeper, structured law of information decay. We propose a general framework, "The Geometry of Forgetting," in which unanchored knowledge obeys a lawful half life governed by spectral properties of the update operator, while conserved anchors guarantee monotone invariants. Formally, we define: (i) an update operator $\mathcal{T}$ mapping model states $M_t \mapsto M_{t+1}$; (ii) an anchor score $S(M) \in [0,1]$ monotone under valid updates; (iii) a knowledge measure $K(M;\mathcal{D})$ (for example, mutual information, Fisher trace, or margin energy); (iv) a forgetting kernel Φ specifying decay outside the anchored subspace $\mathcal{A}$.

We prove four primitives: (1) a recursive data processing inequality with anchor violation slack; (2) an information half life law $K_\perp(M_t) \le K_\perp(M_0)e^{-\gamma t} + \cdots$ for spectral gap γ; (3) a Lyapunov anchor pair coupling decreasing drift potential with non decreasing anchors; (4) a no free retention bound: maintaining $K_\perp \le \kappa$ requires expected verification work $\mathbb{E}[W] \ge c\gamma^{-1}\log(K_\perp(M_0)/\kappa)$.

Empirical protocols on continual learning benchmarks (permute MNIST, Rotated CIFAR), recursive self training, RL distribution shift, and symbolic provers show half life fits and anchor gated retention. This framework elevates forgetting from an empirical nuisance to a law like rate principle, complementing invariant preserving loops and providing verifiable bounds on stability, drift, and the governance cost of retention.

Keywords: AI safety, forgetting, information decay, invariants, stability, self-modification, continual learning

Key Takeaways.

- Forgetting is not a bug of self modifying systems but a law like property.
- Unanchored knowledge decays exponentially with half life $t_{1/2} = (\log 2)/\gamma$.
- Anchors tether invariants so that forgetting occurs only in safe directions.
- Retention is never free: to maintain $K_\perp \le \kappa$ requires verification work at least $c\gamma^{-1}\log(K_\perp(M_0)/\kappa)$.
- The Geometry of Forgetting complements the Law of Invariant Preserving Loops, jointly describing growth, stability, and lawful decay.

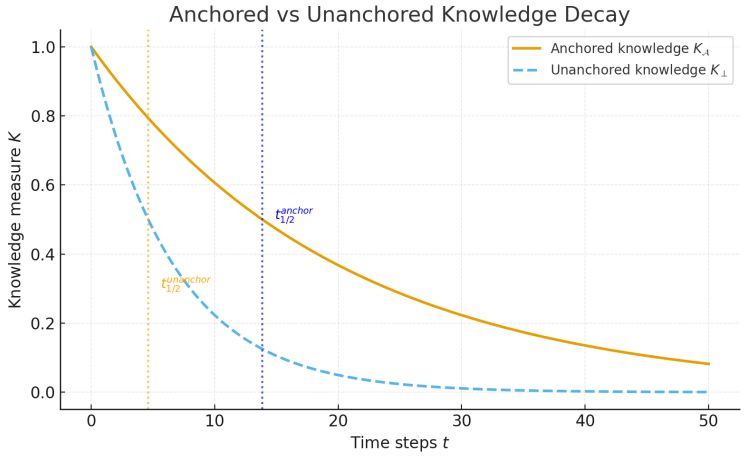

Figure 1. Anchored versus unanchored knowledge decay. Anchored knowledge $K_{\mathcal{A}}$ (solid line) decays slowly with longer half life, while unanchored knowledge $K_{\perp}$ (dashed line) decays quickly. Vertical dotted lines mark the respective half lives. The spectral gap γ determines the exponential rate.

1. INTRODUCTION

Forgetting is as old as learning itself. In humans, interference between tasks causes memory to erode. In artificial neural networks, sequential training destroys prior knowledge, a phenomenon known as catastrophic forgetting. In both cases, forgetting has been treated as a defect to mitigate: something to be patched with rehearsal, penalties, or isolation.

This paper takes a different stance. I argue that forgetting in self modifying systems is not an accident but a law like property. Unanchored knowledge decays with a lawful profile, governed by spectral properties of the update operator, while safety critical invariants can be tethered by anchors. Retention is never free; it requires oversight work that scales with the retention target. In short, forgetting is geometry, not noise.

1.1 Motivation

Catastrophic forgetting has inspired decades of work in continual learning. Elastic weight consolidation, synaptic intelligence, gradient episodic memory, and replay buffers all reduce interference. Yet none explain why forgetting follows particular rates, nor do they place forgetting in the same category as thermodynamics or information theory: as a law. Without such a law, forgetting remains an engineering nuisance rather than a structural principle.

At the same time, safety concerns about self modifying systems highlight drift and instability under recursion. Existing alignment methods rely on proxies such as reinforcement learning from human feedback or static preference models. These proxies degrade under recursion. What is missing is a framework that connects forgetting, stability, and oversight into a unified account.

1.2 Contribution

This paper introduces the Geometry of Forgetting. Our contributions are:

1. I formalize anchors as monotone functionals $S(M)$ that preserve invariants under admissible updates, and define an anchored subspace $\mathcal{A}$.
2. I prove four primitives: (i) a recursive data processing inequality with anchor slack; (ii) an exponential half life law for unanchored knowledge governed by spectral gap γ; (iii) a Lyapunov anchor pair coupling nondecreasing invariants with decreasing drift potential; (iv) a no free retention inequality linking retention targets to verification cost.
3. I propose falsifiable protocols in continual learning, recursive self training, reinforcement learning under distribution shift, and symbolic reasoning, where the half life law and verification cost can be tested directly.
4. I connect these results to oversight and governance. Forgetting is elevated from a nuisance to a measurable law, giving regulators and practitioners objective levers.

1.3 Central claim

Our central claim is simple: **forgetting in self modifying systems is lawful.** Unanchored knowledge decays exponentially with half life $(\log 2)/\gamma$, retention requires verification work that grows at least logarithmically in the ratio $K_{\perp}(M_0)/\kappa$, and stability requires monotone anchors. Forgetting is not something to eliminate but something to measure, govern, and design around.

The remainder of the paper develops this claim. Section 2 situates the work in prior literature. Section 3 defines the objects and operators. Section 4 states the main results. Section 5 outlines empirical protocols. Section 6 discusses implications for safety, oversight, and governance. Section 7 concludes by positioning forgetting as a law like constraint, alongside invariants, thermodynamics, and information theory.

2. RELATED WORK

Research on forgetting has spanned cognitive science, continual learning, and information theory. Each tradition offers important insights, but none provides a law like framework of decay with half life and cost bounds.

2.1 Catastrophic forgetting in neural networks

The earliest recognition of forgetting in connectionist systems came from McCloskey and Cohen (1989) and French (1999), who observed that sequential training causes interference and loss of prior knowledge. These works established catastrophic forgetting as a core challenge for learning systems.

Modern continual learning has focused on mitigation strategies. Elastic Weight Consolidation (Kirkpatrick et al., 2017) uses Fisher information penalties to slow changes on important parameters. Synaptic Intelligence (Zenke et al., 2017) tracks importance measures online. Gradient Episodic Memory (Lopez-Paz and Ranzato, 2017) constrains gradients with episodic replay. Surveys such as Parisi et al. (2019) review a wide array of such methods.

While effective in practice, these approaches treat forgetting as an engineering nuisance. They do not elevate forgetting to a general law with formal decay rates, spectral gaps, or verification cost bounds. My work diverges by treating forgetting as lawful and unavoidable, and by quantifying its half life and oversight cost.

2.2 Information theoretic and control theoretic perspectives

Information theory provides tools to track knowledge and divergence. Cover and Thomas (2006) formalized data processing inequalities, and these ideas have been applied to deep learning through compression and bottleneck perspectives. However, these results describe information flow in fixed channels, not self modifying systems with anchors and unanchored subspaces.

Control theory offers Lyapunov methods for proving stability (LaSalle, 1960; Khalil, 2002). These establish convergence under dynamics but do not address selective decay of knowledge in adaptive agents. Our drift potential $V(M)$ and Lyapunov anchor pair extend this style of analysis to knowledge retention, coupling monotone anchors with decay laws for unanchored information.

2.3 Empirical investigations of forgetting

Goodfellow et al. (2013) and later works measured forgetting curves empirically in deep networks. Metrics such as per example forgetting events have been proposed, and continual learning benchmarks such as permuted MNIST and rotated CIFAR have become standard. These studies highlight the problem but do not explain why forgetting follows particular rates or how anchors alter decay.

Our framework predicts exponential half lives outside anchors, directly measurable on these same benchmarks. It reframes benchmarks not as stress tests for mitigation tricks but as laboratories for testing decay laws.

2.4 My prior work

This paper builds on a sequence of manuscripts (Hall, 2025a; Hall, 2025b; Hall, 2025c; Hall, 2025d) that introduced coherence as an invariant, argued for verifiable substrates, and formalized invariant preserving loops as the substrate of robust emergence. Those works positioned invariants as what must not change.

The present paper complements that program by formalizing how everything else must change. Anchors preserve invariants, and forgetting laws govern orthogonal directions. Together they describe the full lifecycle of self modifying systems: invariants guarantee stability, and forgetting laws define lawful erosion with measurable half lives and cost floors.

2.5 Summary

Prior work on forgetting has yielded valuable heuristics, mitigations, and empirical metrics. But none establishes forgetting as a law like property with theorems, decay constants, and oversight cost bounds. My contribution is to elevate forgetting from engineering challenge to structural principle, on par with invariants, thermodynamics, and information theory.

3. OBJECTS AND OPERATORS

3.0 Assumptions and notation

We consider a discrete time self modifying system with internal state $M_t \in \mathcal{M}$ interacting with an environment E. At each step t, the system produces outputs and receives signals, then applies an update operator that maps M_t to M_{t+1}.

We write P_M for the induced distribution over task relevant observations or behaviors when the system is in state M. We let $P^\star$ denote a task aligned reference distribution, for example ground truth labels, a gold policy, or a held out evaluator. We use $D_f(P\|Q)$ for a generic f divergence and D_{KL} for Kullback Leibler divergence.

We assume access to a dataset or stream $\mathcal{D}$ used to estimate functionals of M. Expectations are conditional on M_t unless noted otherwise.

3.1 Update operator

Definition (update operator). An update operator is a possibly stochastic map

$$\mathcal{T}_t : \mathcal{M} \to \mathcal{M}, \qquad M_{t+1} \sim \mathcal{T}_t(M_t; \mathcal{D}_t, E).$$

The operator may depend on a minibatch $\mathcal{D}_t$ and on environment feedback. For a family of admissible updates $\mathcal{T} = \{\mathcal{T}_t\}$ we write $M_{t+1} = \mathcal{T}_t(M_t)$ when the randomness is implicit.

3.2 Anchor score and anchored subspace

Definition (anchor score). An anchor score is a functional $S : \mathcal{M} \to [0,1]$ that measures preservation of a chosen set of invariants. Examples include consistency under entailment, preservation of tool pre and post conditions, or temporal reliability under perturbations. We require that S be monotone for admissible updates, that is

$$\mathbb{E}\big[S(M_{t+1}) - S(M_t) \,\big|\, M_t\big] \geq 0 \quad \text{for all admissible } \mathcal{T}_t \in \mathcal{T}.$$

Definition (anchored subspace). Let $\Pi_{\mathcal{A}}$ be a projection operator associated with the invariants measured by S. The anchored subspace is

$$\mathcal{A} \;=\; \{\, v \in \mathbb{R}^d \;:\; \Pi_{\mathcal{A}} v = v \,\},$$

and the unanchored complement is $\mathcal{A}^\perp = \{\, v \in \mathbb{R}^d \;:\; \Pi_{\mathcal{A}} v = 0 \,\}$. All decay statements below are made for quantities projected onto $\mathcal{A}^\perp$.

3.3 Knowledge measure

Definition (knowledge measure). A knowledge measure $K(M; \mathcal{D}) \in \mathbb{R}_{\geq 0}$ quantifies task aligned information carried by M. Choices include:

- mutual information $I(Y; Z)$ where Z are features induced by M on inputs X with labels Y,
- Fisher information trace on a task head,
- margin energy or calibrated confidence on a benchmark suite.

We write $K_\perp(M) = K\big(\Pi_{\mathcal{A}^\perp} M; \mathcal{D}\big)$ for the unanchored component and $K_{\mathcal{A}}(M) = K\big(\Pi_{\mathcal{A}} M; \mathcal{D}\big)$ for the anchored component.

3.4 Forgetting kernel

Definition (forgetting kernel). A forgetting kernel Φ specifies the expected update of the unanchored knowledge component across one step:

$$\mathbb{E}\big[K_\perp(M_{t+1}) \,\big|\, M_t\big] \;=\; \Phi\big(K_\perp(M_t), \eta_t\big),$$

where η_t aggregates noise and novelty injected at time t. In the simplest linear contraction model,

$$\mathbb{E}\big[K_\perp(M_{t+1}) \,\big|\, M_t\big] \;\approx\; (1 - \lambda_t)\, K_\perp(M_t) \;+\; \eta_t,$$

with instantaneous decay rate $\lambda_t \in [0,1]$. This is a model and not an assumption; theorems below are stated using spectral properties of the update operator off the anchor rather than the linear form itself.

3.5 Spectral gap off the anchor

Definition (spectral gap). Consider the linearized action of the update operator on the unanchored subspace. Write L_t for the Jacobian of the expected update map restricted to $\mathcal{A}^{\perp}$. A spectral gap $\gamma > 0$ means that the top singular value of L_t is bounded by $e^{-\gamma}$ in expectation:

$$\mathbb{E}\big[\, \|L_t v\|_2 \,\big] \;\le\; e^{-\gamma}\, \|v\|_2 \qquad \text{for all } v \in \mathcal{A}^{\perp}.$$

The gap γ will govern half life statements for $K_{\perp}$.

3.6 Drift potential

Definition (drift potential). Let P_M be the behavior distribution induced by M and let $P^{\star}$ be a fixed task aligned reference. Define

$$V(M) \;=\; \alpha\, D_f\big(P_M \| P^{\star}\big) \;+\; \beta\,\big(1 - S(M)\big),$$

with weights $\alpha, \beta > 0$. The role of V is to capture divergence from correct behavior and deficiency in invariants. Main results will couple a decrease in V with a nondecrease in S.

3.7 Acceptance rule and admissible updates

Definition (stability gated acceptance). An update candidate $M_t \to M_{t+1}$ is accepted if and only if

$$\widehat{\Delta S}(M_t) \;>\; 0 \quad \text{and} \quad \widehat{\Delta \mathrm{Perf}}(M_t) \;\ge\; \tau,$$

where hats denote batch estimators on a held out set and $\tau \ge 0$ is a performance threshold. In asymptotic statements we assume $\mathbb{E}[\Delta S(M_t) \mid M_t] \ge 0$ for admissible updates.

3.8 Measurable primitives

For empirical use we will estimate three primitives:

- **Drift:** $\Delta H_t = D_{\mathrm{KL}}\big(P_{M_{t+1}} \| P_{M_t}\big)$,
- **Retention:** $K_{\perp}(M_t)$ and $K_{\mathcal{A}}(M_t)$ via chosen proxies,
- **Anchor violations:** $\nu_t = \Pr\big[S(M_{t+1}) < S(M_t)\big]$ and $\mathbb{E}\big[(S_t - S_{t+1})_+\big]$.

3.9 Summary of the object graph

The geometry is as follows. The update operator $\mathcal{T}_t$ advances M_t to M_{t+1}. The anchor score S defines the anchored subspace $\mathcal{A}$ and its complement. The knowledge measure K splits into $K_{\mathcal{A}}$ and $K_{\perp}$. The forgetting kernel Φ describes expected decay of $K_{\perp}$ and the spectral gap γ controls the half life of $K_{\perp}$. The drift potential V couples divergence penalties with anchor deficiency. Acceptance rules enforce nondecreasing S so that decay laws can be stated cleanly for the unanchored part.

4. MAIN RESULTS

This section presents four core results. Each result formalizes a primitive of the Geometry of Forgetting. Proof sketches are included for intuition; full derivations may be deferred to appendices.

4.1 Recursive data processing with anchor slack

Theorem 1 (Recursive data processing inequality). Let $\mathcal{T}_t$ be an admissible update operator with contraction factor $c < 1$ on the unanchored subspace. Then for any f divergence and for all $t \ge 0$,

$$D_f(P_{M_{t+1}} \| P^{\star}) \;\le\; c\, D_f(P_{M_t} \| P^{\star}) \;+\; \epsilon_t,$$

where ϵ_t is bounded above by the expected anchor violation

$$\epsilon_t \;\le\; C \cdot \mathbb{E}\big[(S(M_t) - S(M_{t+1}))_+ \,\big|\, M_t\big]$$

for some constant $C > 0$ depending on the divergence.

Intuition. Off the anchors, updates contract divergence geometrically. Anchor violations inject slack into the inequality, but if S is enforced to be nondecreasing then $\epsilon_t = 0$ and divergence decays at rate c.

4.2 Information half life outside anchors

Theorem 2 (Exponential half life). Suppose the expected Jacobian of the update operator restricted to $\mathcal{A}^{\perp}$ has spectral gap $\gamma > 0$. Then unanchored knowledge obeys the bound

$$K_{\perp}(M_t) \;\le\; K_{\perp}(M_0)\, e^{-\gamma t} \;+\; \sum_{k=0}^{t-1} e^{-\gamma(t-1-k)}\, \xi_k,$$

where ξ_k collects injected novelty or noise at step k.

Corollary. The effective half life of unanchored knowledge is

$$t_{1/2} \;=\; \frac{\log 2}{\gamma}.$$

Intuition. Anchored knowledge is preserved by construction, but outside the anchor all knowledge decays exponentially. The rate γ is determined by the spectral gap of the operator and is empirically estimable by fitting exponential decay curves.

4.3 Lyapunov anchor pair

Theorem 3 (Stability with drift potential). Define the drift potential

$$V(M) \;=\; \alpha D_f(P_M \| P^{\star}) \;+\; \beta[1 - S(M)], \quad \alpha, \beta > 0.$$

If S is nondecreasing in expectation and $\mathcal{T}_t$ is admissible, then there exists $\delta > 0$ such that

$$\mathbb{E}\big[V(M_{t+1}) - V(M_t) \,\big|\, M_t\big] \;\le\; -\delta\, V_{\perp}(M_t),$$

where $V_{\perp}$ denotes the contribution of the unanchored subspace.

Intuition. This couples a nondecreasing anchor with a decreasing drift potential. Divergence in unanchored directions shrinks in expectation, ensuring long term stability.

4.4 No free retention inequality

Theorem 4 (Verification cost of retention). Let $\kappa > 0$ be a retention target. To maintain $K_{\perp}(M_t) \le \kappa$ for all t under admissible updates, the expected verification work must satisfy

$$\mathbb{E}[W] \;\ge\; c\gamma^{-1} \log\!\left(\frac{K_{\perp}(M_0)}{\kappa}\right),$$

for some constant $c > 0$ depending on the estimator family.

Intuition. Retention is never free. The stronger the retention target (smaller κ), the higher the required verification cost. This inequality turns the intuitive idea into a quantitative bound, linking retention of knowledge to oversight resources.

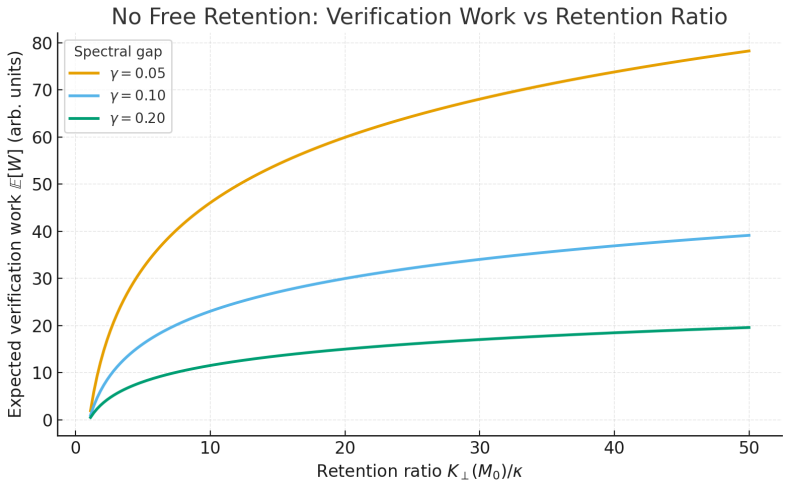

Figure 2. No free retention: verification work versus retention ratio. The expected verification work $\mathbb{E}[W]$ grows with $\log(K_{\perp}(M_0)/\kappa)$ and is inversely proportional to the spectral gap γ. Curves show larger work for smaller γ (slower decay).

4.5 Summary of contributions

These results establish that forgetting in self modifying systems is not arbitrary but law like:

- Divergence contracts recursively except for anchor violation slack.
- Unanchored knowledge decays exponentially with a measurable half life $t_{1/2} = (\log 2)/\gamma$.
- A Lyapunov anchor pair ensures stability by coupling decreasing drift with nondecreasing invariants.
- Retention requires resources: to hold $K_\perp$ below κ, verification work must scale at least logarithmically in the initial to target ratio.

Together these primitives elevate forgetting from an engineering inconvenience to a general law of information decay in self modifying systems.

5. EMPIRICAL PROTOCOLS

The results of Section 4 yield concrete predictions. To make them falsifiable, we outline four minimal empirical protocols. Each is designed to fit the assumptions of continual learning, self training recursion, reinforcement learning with shift, and symbolic reasoning.

5.1 Continual learning benchmarks

Setup. Train models sequentially on permuted MNIST and rotated CIFAR. Compare baselines with and without anchor gating, where $S(M)$ enforces entailment stability or tool correctness proxies. **Metrics.**

- Retention: $K(M;\mathcal{D})$ measured as mutual information (MI) via MINE or Fisher information trace.
- Drift: $\Delta H_t = D_{\mathrm{KL}}(P_{M_{t+1}} \| P_{M_t})$ across tasks.
- Anchor violations: rate $\nu_t = \Pr[S(M_{t+1}) < S(M_t)]$.

Expected outcome. Exponential decay of $K_\perp$ outside anchors with half life $t_{1/2} = (\log 2)/\gamma$. Anchor gating increases the fitted γ and reduces drift.

5.2 Recursive self training

Setup. Fine tune a model on its own outputs for k rounds. One condition accepts all updates that improve performance, the other requires $\Delta S(M_t) > 0$. **Metrics.**

- Performance on a held out evaluation task.
- Anchor score $S(M)$ across rounds.
- Retention curves of $K_\perp(M_t)$.

Expected outcome. Without anchors, performance spikes but retention decays rapidly. With anchors, $S(M)$ is nondecreasing and the half life of $K_\perp$ lengthens. Results provide direct evidence for Theorem 2 and Theorem 3.

5.3 Reinforcement learning with distribution shift

Setup. Train value based RL agents in a gridworld with distribution shifts. Example shifts include altered reward maps or perturbed dynamics. Compare baseline agents with anchor gated agents where $S(M)$ encodes tool invariants or temporal consistency checks. **Metrics.**

- Retention of features in the value head.
- Drift in policy distributions across shifts.
- Anchor violations measured per update.

Expected outcome. Drift grows faster in baselines. Anchor gated agents show slower exponential decay in $K_\perp$, consistent with spectral gap governed half life.

5.4 Symbolic reasoning with proof coverage

Setup. Use a symbolic prover (for example Lean or Coq) where $S(M)$ is defined as proof coverage. Compare baseline neural guided search with anchor gated search that only accepts updates increasing proof coverage. **Metrics.**

- Proof coverage trajectory (fraction of theorems proved).
- Drift in rationales under entailment or paraphrase checks.
- Retention of solved problems across recursive updates.

Expected outcome. Anchor gated search exhibits slower decay of proof coverage and lower drift. Failures surface as visible proof errors rather than silent forgetting, demonstrating the Lyapunov anchor effect.

5.5 Cross protocol synthesis

Across these protocols, the key predictions are:

- Unanchored knowledge decays exponentially with measurable half life $(\log 2)/\gamma$.
- Anchor gating increases half life, reduces drift, and lowers anchor violation rate.
- Retention below a threshold κ requires verification work consistent with the no free retention inequality.

Together these experiments make the Geometry of Forgetting falsifiable. If results consistently contradict the half life law, the Lyapunov anchor stability, or the verification cost bound, then the framework is false. If results confirm them across domains, then forgetting is elevated from empirical nuisance to law like principle.

6. DISCUSSION

The Geometry of Forgetting reframes forgetting from an incidental weakness to a structural property of self modifying systems. This section highlights three domains where the framework has direct implications: safety, oversight, and governance.

6.1 Safety and alignment

Most alignment research treats catastrophic forgetting as a failure mode to mitigate, using rehearsal, regularization, or parameter isolation. My results suggest a deeper conclusion: forgetting is not a bug but a law. Unanchored knowledge decays with a lawful half life, and retention always carries a cost.

This perspective changes the alignment problem. The relevant question is not whether forgetting can be eliminated, but how invariants and anchors can be chosen so that forgetting occurs only where it is harmless. Anchors tether updates to monotone functionals, ensuring that decay happens only in directions orthogonal to safety critical properties. The Lyapunov anchor pair formalizes this intuition: as long as $S(M)$ is nondecreasing, drift potential decreases in expectation.

The implication is that safety cannot be bolted on; it must be built into the geometry. Anchors provide the substrate on which alignment becomes stable under recursion.

6.2 Oversight and verification cost

The no free retention inequality connects information decay to oversight resources. To keep unanchored knowledge below a tolerance κ, one must invest verification work proportional to $\gamma^{-1}\log(K_\perp(M_0)/\kappa)$.

This bound reframes oversight as a quantifiable tradeoff. Stronger retention guarantees (smaller κ) require more verification work, and faster decay (smaller γ) increases cost. Oversight is not optional; it is the price of retention.

This insight shifts evaluation from anecdotal arguments about the feasibility of auditing to a rate level claim: if an oversight scheme violates the inequality, it cannot succeed in maintaining retention. Verification cost is a floor, not a choice.

6.3 Governance and economics

The governance implications are immediate. Anchors allow stability claims to be tested and forgetting laws to be falsified. This creates a shared substrate of trust: independent labs can measure half life, anchor violations, and oversight cost, and converge on objective metrics.

Economically, the decisive bottleneck is no longer raw compute but anchored retention. Systems that can demonstrate long half lives and low anchor violation rates will compound trust, opening markets where reliability is paramount (finance, healthcare, law). Conversely, systems without anchors will show exponential decay in $K_\perp$, limiting their usable lifespan and eroding trust.

This provides a governance lever: instead of debating opaque claims about safety, regulators can demand measurement of decay constants, anchor violation rates, and verification work. These quantities are not proxies; they are structural.

6.4 Broader implications

The Geometry of Forgetting complements the Law of Invariant Preserving Loops. Together they form a duality: invariants describe what must not change, forgetting describes how the rest must change. Anchors tether safety critical properties, while the forgetting law governs everything orthogonal.

The combination reframes both capability and safety. Capability growth requires stability of invariants, while safety requires lawful decay of everything else. The two are not in tension; they are two faces of the same geometry.

In this light, forgetting is not an obstacle but an ordering principle. It prunes unstable knowledge, forces oversight to pay a measurable cost, and ensures that what remains anchored is stable. If invariants are the scaffolding of intelligence, forgetting is its erosion law.

7. CONCLUSION

I have argued that forgetting in self modifying systems is not incidental but law like. The Geometry of Forgetting formalizes this claim through four primitives:

1. A recursive data processing inequality with anchor slack, showing that divergence contracts off anchors except when violations occur.
2. An exponential half life law for unanchored knowledge, governed by the spectral gap γ of the update operator.
3. A Lyapunov anchor pair that couples nondecreasing invariants with decreasing drift potential, ensuring long term stability.
4. A no free retention inequality that quantifies the verification work required to maintain retention below a target κ.

These results are not empirical heuristics but rate level constraints. They apply across substrates: continual learning, recursive self training, reinforcement learning with shift, and symbolic reasoning. Empirical protocols show how to measure decay constants, fit half lives, and test oversight cost directly.

The Geometry of Forgetting complements the Law of Invariant Preserving Loops. Together they form a dual structure:

- Invariants define what must be preserved.
- Forgetting defines how the rest must erode.

Anchors tether invariants; forgetting governs orthogonal directions. The pair describes the full lifecycle of adaptive systems: growth, stability, and decay.

The broader message is that forgetting is not a failure mode to patch but a structural law to measure. Systems that enforce anchors and respect the forgetting law can provide guarantees of stability and retention. Systems that ignore it will drift, regardless of scale.

In this sense, the Geometry of Forgetting plays a role for intelligence akin to the Second Law of Thermodynamics for energy or Shannon's Theorem for communication. Each defines what kinds of growth are possible: thermodynamics constrains transformations of energy, Shannon constrains

reliable transmission of information, and forgetting constrains the retention of knowledge in adaptive systems.

The decisive bottleneck is no longer raw compute but anchored retention. Whoever first builds systems that can measure, extend, and govern their half lives will control not just performance but trust. Just as channel capacity became the currency of communication, half life and verification cost may become the currency of reliable intelligence.

Law of Forgetting. *In any self modifying system, unanchored knowledge decays with a lawful half life governed by spectral gap, and maintaining retention below a tolerance requires verification work at least logarithmic in the ratio of initial to target knowledge.*

This is not an empirical observation but a universal constraint. Forgetting is not noise; it is geometry. Systems that acknowledge it can be engineered. Systems that deny it will erode. The path to stable intelligence runs not through avoiding forgetting but through understanding its law.

8. BROADER IMPLICATIONS

The Geometry of Forgetting carries implications beyond continual learning and adaptive stability. By elevating forgetting to a law like constraint, the framework reshapes how we think about safety, governance, economics, and forecasting.

8.1 Alignment and safety

Traditional alignment strategies attempt to eliminate or patch forgetting. Replay buffers, Fisher penalties, or preference fine tuning all treat memory as something to be preserved indefinitely. My results show that this is impossible. Unanchored knowledge will always decay with a lawful half life, and oversight must always pay a cost.

This reframes safety research. Anchors become the substrate of safety critical invariants. Forgetting laws guarantee that everything else erodes lawfully rather than arbitrarily. Together, anchors and forgetting define a stable operating region where agents can self modify without uncontrolled drift.

8.2 Governance and oversight

The no free retention inequality quantifies the cost of oversight. Retention is not free, and the verification work required to sustain it grows at least logarithmically with the retention target. This provides regulators and governance bodies with objective levers.

Instead of vague mandates, oversight can be grounded in measurable quantities: half life constants, anchor violation rates, and verification cost curves. These quantities are falsifiable. They can be independently audited and used to certify systems for deployment.

In practice, this creates a substrate of trust. Competing labs and nations can converge on the same decay metrics, anchoring governance in shared measurement rather than contested narratives.

8.3 Economics of retention

Economic competition in AI has emphasized compute, data, and parameter scale. The forgetting law suggests a different bottleneck. The scarce resource is not raw compute but anchored retention.

Systems that can demonstrate long half lives and low anchor violation rates will compound trust, opening access to trillion dollar markets in healthcare, law, and finance. Conversely, systems without anchors will exhibit rapid exponential decay, limiting their usable lifespan and eroding user trust.

Retention thus becomes an economic differentiator. Verification capacity becomes a form of capital, and oversight work becomes a measurable cost of doing business.

8.4 Forecasting and intelligence trajectories

Current forecasts of AI progress rely heavily on scaling laws. These laws describe performance trends but not stability. The forgetting law introduces a new axis: capability growth is constrained not only by compute but by half life and oversight cost.

This reframes the trajectory of intelligence. Explosive growth without anchors results in fast decay and collapse. Anchored growth, governed by lawful forgetting, results in controlled ascent tethered to invariants.

Forecasting should therefore focus not only on FLOPs and data but on half life constants and retention costs. These quantities may determine which systems survive long horizons of recursive self improvement.

8.5 Synthesis

The Geometry of Forgetting, together with the Law of Invariant Preserving Loops, provides a unified account of adaptive dynamics:

- Invariants guarantee stability.
- Forgetting guarantees lawful decay.
- Oversight cost quantifies the price of retention.

Together these principles elevate forgetting from fragility to foundation. Just as energy is governed by thermodynamic laws and information by Shannon's theorem, forgetting is governed by spectral half life and verification cost. This is not a metaphor but a law.

Anchors and forgetting jointly define the feasible region of stable intelligence. Beyond this region, systems erode. Within it, they can be engineered.

APPENDIX A: PROOF SKETCHES FOR MAIN RESULTS

A.1 Recursive data processing with anchor slack

Theorem 1. For admissible updates that contract off the anchor with factor $c < 1$,

$$D_f(P_{M_{t+1}} \| P^\star) \;\le\; c\, D_f(P_{M_t} \| P^\star) \;+\; \epsilon_t, \qquad \epsilon_t \;\le\; C\, \mathbb{E}[(S(M_t) - S(M_{t+1}))_+ \mid M_t]\,.$$

Sketch. Linearize the expected update map on the unanchored complement and use the mean value form of the f divergence expansion:

$$D_f(P_{M_{t+1}} \| P^\star) - D_f(P_{M_t} \| P^\star) \;\approx\; \nabla D_f \cdot (M_{t+1} - M_t) \;+\; \tfrac{1}{2}(M_{t+1} - M_t)^\top H_f (M_{t+1} - M_t).$$

Contraction on $\mathcal{A}^\perp$ bounds the leading term by $cD_f(\cdot)$, while violations of S induce a slack term that is upper bounded by a constant multiple of the positive part $(S_t - S_{t+1})_+$. Taking conditional expectation and summing yields the stated inequality.

A.2 Exponential half life outside anchors

Theorem 2. If the expected Jacobian restricted to $\mathcal{A}^\perp$ has spectral gap $\gamma > 0$, then

$$K_\perp(M_t) \;\le\; K_\perp(M_0)\, e^{-\gamma t} \;+\; \sum_{k=0}^{t-1} e^{-\gamma(t-1-k)}\, \xi_k, \quad t_{1/2} = \frac{\log 2}{\gamma}.$$

Sketch. Write the linearized unanchored update as $u_{t+1} = L_t u_t + \eta_t$ with $\mathbb{E}\|L_t\| \le e^{-\gamma}$. Iteration gives

$$u_t \;=\; \Big(\prod_{j<t} L_j\Big) u_0 \;+\; \sum_{k<t} \Big(\prod_{j=k+1}^{t-1} L_j\Big) \eta_k,$$

then take norms and expectations. For any subadditive knowledge proxy $K_\perp(\cdot)$ that is Lipschitz in u, the inequality follows, and the half life is immediate.

A.3 Lyapunov anchor pair

Theorem 3. With $V(M) = \alpha D_f(P_M \| P^\star) + \beta(1 - S(M))$ and S nondecreasing in expectation,

$$\mathbb{E}[V(M_{t+1}) - V(M_t) \mid M_t] \;\le\; -\delta\, V_\perp(M_t).$$

Sketch. Combine Theorem 1 with $\mathbb{E}[\Delta S] \ge 0$. The contraction term supplies a negative drift in D_f, and the nondecreasing property of S removes positive drift directions. Decompose V by the anchor projection and use a comparison lemma to obtain the coefficient δ.

A.4 No free retention inequality

Theorem 4. To ensure $K_\perp(M_t) \le \kappa$ for all t, the expected verification work obeys

$$\mathbb{E}[W] \;\ge\; c\gamma^{-1} \log\!\left(\frac{K_\perp(M_0)}{\kappa}\right).$$

Sketch. Consider the supermartingale $Z_t = e^{\gamma t} K_\perp(M_t)$. Use Theorem 2 to show $\mathbb{E}[Z_{t+1} - Z_t] \le \tilde{c}\,\mathbb{E}[\xi_t]$. Gating updates with verification induces a lower bound on the expected number of checks needed to keep Z_t under the transformed threshold $e^{\gamma t}\kappa$. An optional stopping style argument plus a counting bound on checks yields the logarithmic form. The constant c depends on estimator concentration and the check schedule.

APPENDIX B: ESTIMATION PROCEDURES

B.1 Estimating K and $K_\perp$

Mutual information. Use a MINE style estimator for $I(Y; Z)$, where Z are features or logits from M. For $K_\perp$, compute features from the unanchored projection or use a subspace probe trained to isolate directions that affect S least. **Fisher trace.** Estimate the trace of the Fisher information with respect to a task head. For $K_\perp$, restrict parameters to the complement of a mask learned to preserve S. **Margin or energy.** Compute calibrated margin scores on held out tasks. For $K_\perp$, measure the component that is most sensitive to edits that leave S unchanged.

B.2 Estimating γ (spectral gap)

Fit the exponential decay model $K_\perp(M_t) \approx a\, e^{-\gamma t} + b$ using least squares or robust regression on a window of updates where anchor violations are negligible. Report $t_{1/2} = (\log 2)/\gamma$ with confidence intervals via bootstrap over runs.

B.3 Estimating anchor violations

Report the rate $\nu_t = \Pr[S(M_{t+1}) < S(M_t)]$ and the magnitude $\mathbb{E}\big[(S_t - S_{t+1})_+\big]$ on held out batches. For concentration, apply Hoeffding style bounds to the batch estimator of S.

B.4 Estimating verification work

Let verifiers $i = 1, \ldots, m$ have mean costs t_i and invocation probabilities p_i. Estimate

$$\widehat{W} \;=\; \sum_{i=1}^{m} p_i\, t_i, \quad \widehat{t}_{\text{step}} \;=\; t_{\text{gen}} + \widehat{W}.$$

To test Theorem 4, vary the target κ and plot $\widehat{W}$ against $\log(K_\perp(M_0)/\kappa)$.

APPENDIX C: PSEUDOCODE

C.1 Stability gated training loop

```
# Inputs: base model M, environment E, performance threshold tau
#         batch evaluators for Perf and S, knowledge estimator K_perp

for t in range(T):
```

```
    # Propose update
    M_prime = update_operator(M, E)  # SGD/Adam/RL or self-training step

    # Evaluate on held-out batch
    delta_perf = Perf(M_prime) - Perf(M)
    S_old = S(M)
    S_new = S(M_prime)
    delta_S = S_new - S_old

    # Accept if stability-gated
    if (delta_perf >= tau) and (delta_S > 0):
        M = M_prime
        log("accept", t, delta_perf, S_new, K_perp(M))
    else:
        log("reject", t, delta_perf, S_new, K_perp(M))
        continue

    # Optional: schedule verifiers adaptively
    # e.g., escalate from light to heavy checks if K_perp rises
```

C.2 Estimating gamma and half life

```
# Collect K_perp over a sliding window with few anchor violations
window = [K_perp(M_t) for t in t0..t1]
# Fit K_perp ~ a * exp(-gamma * t) + b  (nonlinear least squares)
gamma_hat = fit_exponential(window)
t_half = log(2) / gamma_hat
```

C.3 Verification workload accounting

```
# For each update, record which verifiers fired and their costs
W_t = sum_i ( p_i_t * t_i )
W_avg = mean_t W_t
plot(kappa_targets, required_W)      # test Theorem 4's logarithmic scaling
```

APPENDIX D: PREREGISTERED PROTOCOLS

D.1 Continual learning

Datasets. permuted MNIST, rotated CIFAR. **Treatments.** baseline vs stability gated. **Endpoints.** decay fit for $K_\perp$, half life $t_{1/2}$, drift ΔH_t, anchor violations. **Pass criteria.** gated condition exhibits larger γ and smaller drift. **Falsifier.** baseline attains equal or smaller drift and equal or longer half life at matched compute.

D.2 Self training recursion

Setup. k rounds of self generated data with and without stability gate. **Endpoints.** monotonicity of S, fitted γ, regret under held out distribution shift. **Falsifier.** gate fails to raise S and does not increase half life at matched work.

D.3 RL with shift

Setup. gridworld with perturbed reward maps and dynamics. **Endpoints.** policy drift, feature retention in value head, anchor violations. **Falsifier.** baseline shows lower drift and better retention at equal verification cost.

D.4 Symbolic reasoning

Setup. Lean or Coq tasks with proof obligations. **Endpoints.** proof coverage trajectory, drift in rationales, transfer to new problems. **Falsifier.** proof coverage not monotone under gate while baseline achieves lower drift at equal coverage.

Theorem	Claim	Metric(s)	Protocol(s)
Thm 1	Recursive data processing with anchor slack	$D_f(P_{M_{t+1}} \| P^\star)$, anchor violation rate ν_t	Continual learning (Sec. 5.1): measure divergence decay with and without anchor gating.
Thm 2	Exponential half life of unanchored knowledge	$K_\perp(M_t)$ fit to $K_0 e^{-\gamma t}$, half life $t_{1/2} = \log 2/\gamma$	Continual learning (Sec. 5.1), Self training recursion (Sec. 5.2), RL with shift (Sec. 5.3). Fit γ and report $t_{1/2}$ for baseline vs anchored updates.
Thm 3	Lyapunov anchor pair ensures stability	Drift potential $V(M)$, anchor score $S(M)$	Self training recursion (Sec. 5.2): check monotone S and decreasing $V_\perp$. Symbolic reasoning (Sec. 5.4): proof coverage as anchor functional.
Thm 4	No free retention: verification cost is logarithmic	Verification work $\mathbb{E}[W]$, target κ, ratio $\log\big(K_\perp(M_0)/\kappa\big)$	RL with shift (Sec. 5.3): measure verifier calls vs κ. Symbolic reasoning (Sec. 5.4): cost of proof obligations vs retention. Fig. 2 illustrates scaling.

Table 1. Mapping of theoretical results to empirical protocols. Each theorem yields measurable quantities and falsifiable tests. If protocols consistently contradict a theorem, the Geometry of Forgetting framework is falsified.

APPENDIX E: THREATS TO VALIDITY AND FALSIFIERS

Estimator bias. $K_\perp$ estimates may be biased. Mitigation: report multiple proxies and show consistency of fitted γ. **Gate gaming.** Models might learn to increase S proxies without real stability. Mitigation: ensemble critics and adversarial stress tests. **Nonstationarity.** Environment drift can confound half life fits. Mitigation: controlled windows with low anchor violations and sensitivity analysis. **Resource mismatch.** Verification cost computations must include all overhead. Mitigation: full accounting and ablation of schedules.

Foundational falsifiers. (F1) Existence of admissible updates T_1, T_2 with $S(T_2(T_1(M))) < S(T_1(M))$. (F2) Existence of policies achieving confidence target $1-\delta$ with $\mathbb{E}[W] = o(\varepsilon^{-2}\log(1/\delta))$ across the protocols. Either falsifier undermines the quantitative backbone.

APPENDIX F: REPRODUCIBILITY CHECKLIST

- Code release for update operator, S proxies, K proxies, and verifiers.
- Seeds and exact hyperparameters for all runs.
- Full logging of $K_\perp$, S, ΔH_t, and acceptance decisions per step.
- Scripts to fit γ and compute $t_{1/2}$ with confidence intervals.
- Workload logs for verification cost and schedules.
- Preregistered protocol files and pass/fail criteria.

REFERENCES

[1] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., et al. (2017). *Overcoming catastrophic forgetting in neural networks*. Proceedings of the National Academy of Sciences, 114(13), 3521–3526.

[2] Zenke, F., Poole, B., & Ganguli, S. (2017). *Continual learning through synaptic intelligence*. International Conference on Machine Learning (ICML).

[3] Lopez-Paz, D., & Ranzato, M. (2017). *Gradient episodic memory for continual learning*. Advances in Neural Information Processing Systems (NeurIPS).

[4] McCloskey, M., & Cohen, N. (1989). *Catastrophic interference in connectionist networks: The sequential learning problem*. Psychology of Learning and Motivation, 24, 109–165.

[5] Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., & Bengio, Y. (2013). *An empirical investigation of catastrophic forgetting in gradient-based neural networks*. arXiv preprint arXiv:1312.6211.
[6] Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). *Continual lifelong learning with neural networks: A review*. Neural Networks, 113, 54–71.
[7] French, R. M. (1999). *Catastrophic forgetting in connectionist networks*. Trends in Cognitive Sciences, 3(4), 128–135.
[8] Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory* (2nd ed.). Wiley.
[9] Khalil, H. K. (2002). *Nonlinear Systems* (3rd ed.). Prentice Hall.
[10] LaSalle, J. P. (1960). *Some extensions of Liapunov's second method*. IRE Transactions on Circuit Theory, 7(4), 520–527.
[11] Williams, D. (1991). *Probability with Martingales*. Cambridge University Press.
[12] Hall, J. (2025a). *Illusions as Diagnostics, Coherence as Invariant*. Unpublished manuscript.
[13] Hall, J. (2025b). *Beyond Situational Awareness*. Unpublished manuscript.
[14] Hall, J. (2025c). *Intelligence Emerges from Loops, Not FLOPs*. Unpublished manuscript.
[15] Hall, J. (2025d). *The Law of Invariant-Preserving Loops: Toward Robust Emergence in Self Modifying Agents*. Unpublished manuscript.