The Illusion of State: Sharp Memory-Decay Bounds in Linear State Space Models

Aryan Dadwal [1]

Abstract

Linear State Space Models (SSMs) have emerged as a powerful alternative to attention for long-sequence modeling. The prevailing assumption is that memory decays "exponentially" when the transition matrix is stable. We show this characterization is incomplete. By analyzing the Jordan normal form of A, we prove that memory decay follows $\Theta(k^{m-1}\rho^k)$, where ρ is the spectral radius and m is the size of the dominant Jordan block. We provide matching upper and lower bounds, extend the analysis to the input-output map via controllability and observability Gramians, and generalize to time-varying gated architectures. Empirically, we discover a fundamental architectural tradeoff: defective (non-diagonalizable) transition matrices drastically extend memory retention on associative recall—even when learned via gradient descent—but degrade performance on formal language tasks requiring independent state counting. LayerNorm resolves the numerical explosion caused by polynomial transients while preserving the memory advantage. Our results provide the first sharp characterization of the effective context horizon in SSMs and offer concrete design principles for practitioners.

1. Introduction

Linear State Space Models—including S4 (Gu et al., 2022), DSS (Gupta et al., 2022), and Mamba (Gu & Dao, 2023)—have established themselves as competitive models for long-sequence tasks, computing a hidden state recursively via $h_{t+1} = Ah_t + Bx_t$. The community widely accepts that if A is stable ($\rho(A) < 1$), memory decays exponentially. However, "exponential decay" is a loose heuristic. Transition matrices in modern SSMs are often non-normal and may be highly defective. In these regimes, transient amplification and polynomial delay factors dominate the exponential envelope over the effective context window, fundamentally altering the memory landscape. Merrill et al. (Merrill et al., 2024) prove that linear SSMs are restricted to TC^0, characterizing exact long-range state tracking as an "illusion." Sarrof et al. (Sarrof et al., 2024) show the expressive capacity is bottlenecked by algebraic structure. Our work provides the analytic counterpart: we quantify the *rate* of state leakage via $\Theta(k^{m-1}\rho^k)$ and show how expressive bottlenecks correspond to unobservable modes in the Jordan structure. Classical control theory (Kailath, 1980) establishes connections between Gramians and observability; we extend these to provide *sharp*, constructive bounds on modern SSM architectures.

Contributions. (1) We derive a provably tight bound $\Theta(k^{m-1}\rho^k)$ incorporating the polynomial prefactor from the Jordan structure of A. (2) We extend this to the input-output map, proving the polynomial prefactor is gated by controllability and observability Gramians. (3) We generalize to time-varying gated architectures (Mamba (Gu & Dao, 2023), Griffin (De et al., 2024)). (4) We empirically validate a fundamental tradeoff: defectiveness extends memory recall but degrades formal language counting, and LayerNorm resolves the associated numerical instability.

2. Theoretical Bounds

2.1. Preliminaries

We consider the discrete-time linear SSM:

$$h_{t+1} = Ah_t + Bx_t, \qquad y_t = Ch_t, \tag{1}$$

where $h_t \in \mathbb{C}^n$, $x_t \in \mathbb{C}^d$, $y_t \in \mathbb{C}^p$. We assume strict stability: $\rho(A) < 1$. With $h_0 = 0$, the state is $h_t = \sum_{k=1}^{t} A^{k-1}Bx_{t-k}$. We define the **state memory** $\mathcal{M}_h(k) = \|A^{k-1}B\|$ and **input-output memory** $\mathcal{M}_y(k) = \|CA^{k-1}B\|$.

Every A admits a Jordan decomposition $A = PJP^{-1}$, where $J = \mathrm{blockdiag}(J_1, \ldots, J_q)$ and each $J_i = \lambda_i I_{m_i} + N_{m_i}$, with N_{m_i} nilpotent of degree m_i. A *dominant Jordan*

[1]Indian Institute of Technology, Jodhpur, India. Correspondence to: Aryan Dadwal <b24bs1070@iitj.ac.in>.

Proceedings of the DEMO Workshop at the 43^{rd} International Conference on Machine Learning (DEMO@ICML 2026), Seoul, South Korea. 2026.

block has $|\lambda_i| = \rho(A)$ and maximal size m.

2.2. Sharp State Memory Bound

Theorem 2.1 (Sharp state memory)**.** *For any strictly stable linear SSM with transition matrix A, spectral radius ρ, and dominant Jordan block size m:*

$$\|A^k\| = \Theta(k^{m-1}\rho^k). \tag{2}$$

The upper bound follows from $A^k = PJ^kP^{-1}$ and the binomial expansion of each Jordan block: $J_i^k = \sum_{j=0}^{m_i-1} \binom{k}{j} \lambda_i^{k-j} N^j$, with dominant term $\binom{k}{m-1}\lambda^{k-m+1}$, bounded by $k^{m-1}\rho^k/(m-1)!$ up to constants. The matching lower bound extracts the $(1,m)$-entry of the dominant block via Cauchy–Schwarz: $\|A^k\| \geq |q_1^H A^k p_m|/(\|q_1\|\|p_m\|)$, where p_m and q_1 are the corresponding generalized eigenvectors (full proofs in Section A).

Effective memory horizon. Maximizing $f(k) = k^{m-1}\rho^k$ yields a closed-form peak location:

$$k_{\max} = \frac{m-1}{-\ln \rho} \approx \frac{m-1}{1-\rho} \quad \text{for } \rho \text{ near } 1. \tag{3}$$

For a diagonal matrix ($m = 1$), the peak is at $k = 0$—pure exponential decay with no transient phase. For a 5×5 Jordan block with $\rho = 0.99$, the peak shifts to $k \approx 400$, creating a massive effective context window.

2.3. Input-Output Memory and Control Geometry

While Theorem 2.1 bounds intrinsic memory, the observed memory $\mathcal{M}_y(k)$ depends on how inputs excite the state (B) and how outputs read from it (C).

Definition 2.2 (Effective block size)**.** For Jordan block J_i of size m_i, the effective block size w.r.t. (B, C) is:

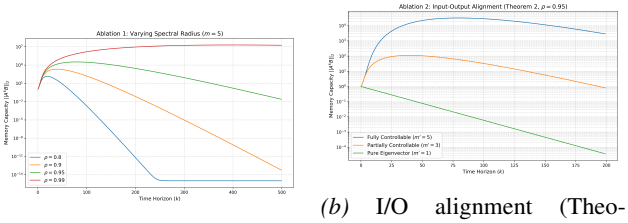
$$m_i' = \max\{j{+}1 : 0 \leq j \leq m_i{-}1,\ \tilde{C}_i N^j \tilde{B}_i \neq 0\}, \tag{4}$$

where $\tilde{C} = CP$, $\tilde{B} = P^{-1}B$ are projections in the Jordan basis.

Theorem 2.3 (Sharp input-output memory)**.** *Let* $\rho_{\text{eff}} = \max\{|\lambda_i| : m_i' > 0\}$ *and* $d_{\text{eff}} = \max\{m_i'{-}1 : |\lambda_i| = \rho_{\text{eff}}\}$. *Then:*

$$\|CA^kB\| = \Theta\big(k^{d_{\text{eff}}}\,\rho_{\text{eff}}^k\big). \tag{5}$$

The conditions for $m_i' < m_i$ are encoded by the controllability and observability Gramians. If B and C lie in subspaces orthogonal to the dominant generalized eigenvectors, the corresponding Gramian blocks vanish, strictly accelerating memory decay.



(a) Varying ρ (Theorem 2.1). *(b)* I/O alignment (Theorem 2.3).

Figure 1. Synthetic validation. (a) Peak of $k^{m-1}\rho^k$ shifts as $k_{\max} \approx (m{-}1)/(1{-}\rho)$. (b) Structural uncontrollability truncates the polynomial prefactor.

2.4. Time-Varying Gated Architectures

Modern architectures like Mamba construct input-dependent transitions $A_\tau = \exp(-\Delta_\tau) \odot A$, replacing A^k with $\prod_{\tau=1}^k A_\tau$. The gating squashes $\bar{\rho} = \mathbb{E}[\|A_\tau\|] < 1$. By sub-multiplicativity and Jensen's inequality, the expected memory envelope remains $\mathcal{O}(k^{m-1}\bar{\rho}^k)$. Because gating compresses $\bar{\rho}$, **the polynomial prefactor is the principal structural mechanism to delay forgetting** in time-varying architectures.

3. Experiments

We validate our bounds through synthetic ablations and neural network training on discrete sequence tasks. All experiments run on a single CPU ($\leq$6 GB RAM).

3.1. Synthetic Validation

Varying ρ. We fix $m = 5$ and vary $\rho \in \{0.8, 0.9, 0.95, 0.99\}$. Figure 1(a) confirms the peak of $k^{m-1}\rho^k$ shifts as $k_{\max} \approx (m{-}1)/(1{-}\rho)$: from $k \approx 20$ at $\rho = 0.8$ to $k \approx 400$ at $\rho = 0.99$.

Input-output alignment. We fix $m = 5$, $\rho = 0.95$ and set B to achieve effective sizes $m' \in \{5, 3, 1\}$. Figure 1(b) shows the polynomial degree drops exactly as predicted by Theorem 2.3.

Algebraic vs. geometric multiplicity. We compare a single Jordan block ($m = 5$) against a diagonal matrix with 5 repeated eigenvalues (Figure 2). Only the defective matrix exhibits polynomial transients, confirming that defectiveness—not merely repeated eigenvalues—drives the phenomenon.

3.2. The Polynomial Explosion and Its Resolution

We train lightweight SSMs ($d_{\text{model}} = 16$, $\rho = 0.9$) on continuous associative recall. Without normalization, the defective matrix causes hidden states to reach $\sim 10^{26}$, making training impossible—explaining why S4 enforces diagonalizability. Applying LayerNorm to the hidden state at each step completely resolves the explosion. The m=16 + Layer-

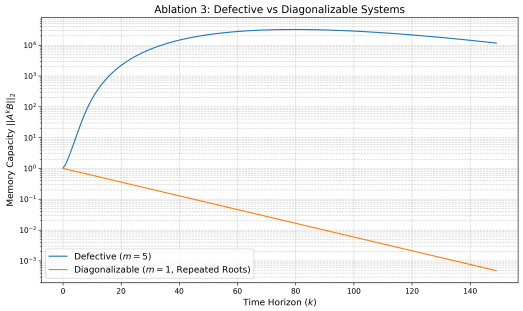
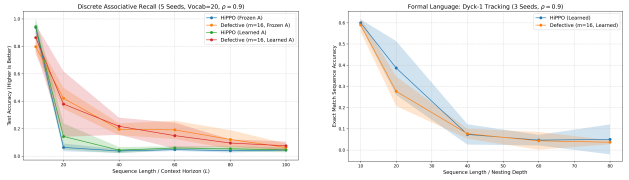


Figure 2. Algebraic vs. geometric multiplicity. Only the defective (Jordan block) matrix exhibits polynomial transients; the diagonal matrix with repeated eigenvalues decays purely exponentially.



(a) Discrete recall (5 seeds). *(b)* Dyck-1 language (3 seeds).

Figure 3. The fundamental tradeoff. (a) Defective + LayerNorm dominates HiPPO on memory recall. (b) HiPPO outperforms on Dyck-1 counting, because diagonal structure preserves independent state tracking. Shaded: ± 1 std.

Norm model achieves the lowest MSE across all sequence lengths (0.28 at L=60 vs. HiPPO's 0.84). LayerNorm preserves the *direction* of the hidden state while rescaling its magnitude; the readout C can still decode relative component proportions encoding memory age and content.

3.3. The Fundamental Tradeoff

We use discrete vocabulary (V=20), embedding layers, cross-entropy loss, and 5 random seeds.

Discrete associative recall (Figure 3a). The Defective + LayerNorm initialization statistically dominates HiPPO (Gu et al., 2020): at L=40, HiPPO drops to chance ($1/V = 0.05$), while the defective model retains $\sim$0.20 accuracy. This holds even when A is learned via gradient descent.

Dyck-1 formal language (Figure 3b). On balanced-parenthesis tracking, the defective model performs worse than HiPPO (0.28 vs. 0.39 exact-match at L=20). The mechanism is clear: Dyck-1 requires independent state counting. A diagonal matrix keeps dimensions independent, acting as parallel counters. A defective matrix has ones on the superdiagonal, mixing dimensions via $h_{i,t} = \lambda h_{i,t-1} + h_{i+1,t-1}$—precisely the mechanism that delays decay but destroys counting capacity.

4. Discussion and Conclusion

Design principle. Our results reveal a fundamental tradeoff: Jordan blocks trade independent state capacity (needed for counting) for polynomial memory extension (needed for recall). Practitioners can leverage this with *mixed* initializations—diagonal blocks for counting-dependent layers, small defective blocks ($m = 2$–4) for memory-critical layers.

Explaining existing architectures. Our theory explains why S4 enforces HiPPO initialization: it avoids polynomial explosion at the cost of shorter effective context. It also explains why Mamba's gating squashes $\bar{\rho}$, making defectiveness the only remaining structural lever for extending memory.

Limitations. Our bounds assume fixed Jordan structure; gradient descent may alter eigenstructure during training. Our models are lightweight ($d_{\text{model}} = 16$); scaling to production-size SSMs is left to future work. The time-varying analysis provides expected rather than worst-case bounds.

Conclusion. Memory decay in linear SSMs follows $\Theta(k^{m-1}\rho^k)$, not pure exponential decay. The polynomial prefactor from Jordan block defectiveness creates a massive transient phase that can be harnessed—via LayerNorm or careful block sizing—to extend effective context far beyond diagonalizable initializations. This extension comes at the cost of independent state counting capacity, revealing a fundamental tradeoff with concrete implications for SSM architecture design.

Impact Statement

This paper provides theoretical characterizations of memory in linear sequence models. The results are foundational and carry no foreseeable negative societal consequences beyond those generally associated with advances in sequence modeling.

References

De, S., Smith, S. L., Fernando, A., et al. Griffin: Mixing gated linear recurrences with local attention for efficient language models. *arXiv preprint arXiv:2402.19427*, 2024.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Gu, A., Dao, T., Ermon, S., Rudra, A., and Ré, C. HiPPO: Recurrent memory with optimal polynomial projections. In *Advances in Neural Information Processing Systems*, 2020.

Gu, A., Goel, K., and Ré, C. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Gupta, A., Gu, A., and Berant, J. Diagonal state spaces are as effective as dense state spaces. *arXiv preprint arXiv:2203.14343*, 2022.

Kailath, T. *Linear Systems*. Prentice-Hall, 1980.

Merrill, W., Petty, J., and Sabharwal, A. The illusion of state in state-space models. In *International Conference on Machine Learning*, 2024.

Sarrof, Y., Veitsman, Y., and Hahn, M. The expressive capacity of state space models: A formal language perspective. In *Advances in Neural Information Processing Systems*, 2024.

A. Full Proofs

A.1. Proof of Upper Bound (Theorem 2.1, upper direction)

By the Jordan decomposition $A = PJP^{-1}$ and sub-multiplicativity:

$$\|A^k\| \le \kappa(P) \max_i \|J_i^k\|, \quad \kappa(P) = \|P\|\|P^{-1}\|. \tag{6}$$

For each block, $J_i^k = \sum_{j=0}^{m_i-1} \binom{k}{j} \lambda_i^{k-j} N_{m_i}^j$. Using $\binom{k}{j} \le k^j/j!$ and $\|N_{m_i}^j\| \le 1$:

$$\|J_i^k\|_\infty \le \sum_{j=0}^{m_i-1} \frac{k^j}{j!} |\lambda_i|^{k-j} \le m_i \cdot \frac{k^{m_i-1}}{(m_i-1)!} \cdot \rho^{k-m_i+1}. \tag{7}$$

Taking the maximum over all blocks and setting $C_U = \kappa(P) \cdot n \cdot \rho^{-(m-1)}/(m-1)!$ yields $\|A^k\| \le C_U k^{m-1} \rho^k$ for all $k \ge m-1$. □

A.2. Proof of Lower Bound (Theorem 2.1, lower direction)

We extract the $(1,m)$-entry of the dominant block. Let p_m be the m-th column of P (the highest-grade generalized eigenvector corresponding to the dominant block) and q_1^H be the first row of P^{-1} (the left eigenvector):

$$q_1^H A^k p_m = (J_{\text{dom}}^k)_{1,m} = \binom{k}{m-1} \lambda^{k-m+1}. \tag{8}$$

By Cauchy–Schwarz:

$$\|A^k\|_2 \ge \frac{|q_1^H A^k p_m|}{\|q_1\|_2 \|p_m\|_2}. \tag{9}$$

Since $\binom{k}{m-1} \sim k^{m-1}/(m-1)!$ for large k, the lower bound follows with constant $C_L = (\|q_1\|_2 \|p_m\|_2 (m-1)! \rho^{m-1})^{-1}$. □

A.3. Proof of Theorem 2.3 (Input-Output Memory)

Working in the Jordan basis, $CA^kB = \tilde{C} J^k \tilde{B}$ where $\tilde{C} = CP$ and $\tilde{B} = P^{-1}B$. For each block:

$$\tilde{C}_i J_i^k \tilde{B}_i = \sum_{j=0}^{m_i-1} \binom{k}{j} \lambda_i^{k-j} \, \tilde{C}_i N_{m_i}^j \tilde{B}_i. \tag{10}$$

The highest-order non-vanishing term has $j = m_i' - 1$ (by definition of effective block size). The product $\tilde{C}_i N^{m_i-1} \tilde{B}_i$ reduces to the outer product of the first column of $\tilde{C}_i$ and the last row of $\tilde{B}_i$; if either vanishes, the mode is structurally unobservable or uncontrollable, and $m_i' < m_i$.

Taking the maximum over blocks with $m_i' > 0$ and applying the same upper/lower bound technique as in Theorem 2.1 yields $\|CA^kB\| = \Theta(k^{d_{\text{eff}}} \rho_{\text{eff}}^k)$. □

B. Effective Memory Horizon Derivation

The memory envelope $f(k) = k^{m-1}\rho^k$ achieves its maximum where $f'(k) = 0$. Taking the logarithmic derivative:

$$\frac{d}{dk} \ln f(k) = \frac{m-1}{k} + \ln \rho = 0 \implies k_{\max} = \frac{m-1}{-\ln \rho}. \tag{11}$$

For ρ near 1, $-\ln \rho \approx 1 - \rho$, giving $k_{\max} \approx (m-1)/(1-\rho)$. This has clear design implications:

- **Diagonal matrices** ($m = 1$): $k_{\max} = 0$. No transient phase; pure exponential decay from $k = 0$.
- **HiPPO initialization** ($m = 1$, diagonalizable): Same as diagonal. Memory is extended only through spectral radius ρ being close to 1.
- **Defective block** ($m = 5$, $\rho = 0.99$): $k_{\max} \approx 400$. The polynomial prefactor creates a 400-step window during which memory actually *increases* before exponential decay takes over.

C. Gramian Interpretation

The controllability and observability Gramians W_c and W_o solve the discrete Lyapunov equations:

$$AW_cA^H - W_c + BB^H = 0, \tag{12}$$

$$A^HW_oA - W_o + C^HC = 0. \tag{13}$$

In the Jordan basis, these have block structure. The (i, j)-block of $\tilde{W}_c = P^{-1}W_cP^{-H}$ satisfies a Sylvester equation involving J_i and J_j^H. For the diagonal blocks, $(\tilde{W}_c)_{ii}$ encodes which rows of $\tilde{B}_i$ are nonzero—precisely the information determining m_i'.

The effective block size m_i' can be read off from the rank structure of the corresponding Gramian blocks:

- If $(\tilde{W}_c)_{ii}$ has zero last row: the highest generalized eigenvector is uncontrollable, and $m_i' < m_i$.
- If $(\tilde{W}_o)_{ii}$ has zero first column: the lowest generalized eigenvector is unobservable, and $m_i' < m_i$.

This provides a numerically stable way to assess the effective memory order without computing the Jordan decomposition explicitly.

D. Time-Varying Analysis Details

For input-dependent transitions $A_\tau = \exp(-\Delta_\tau) \odot A$ (as in Mamba), the state at time t involves the product $\prod_{\tau=1}^{k} A_\tau$. By sub-multiplicativity:

$$\Big\| \prod_{\tau=1}^{k} A_\tau \Big\| \le \prod_{\tau=1}^{k} \|A_\tau\|. \tag{14}$$

Taking logarithms and applying Jensen's inequality:

$$\frac{1}{k} \sum_{\tau=1}^{k} \ln \|A_\tau\| \le \ln \mathbb{E}[\|A_\tau\|] = \ln \bar{\rho}. \tag{15}$$

The expected memory envelope is thus $\mathcal{O}(k^{m-1}\bar{\rho}^k)$. Because the non-negative gating $\exp(-\Delta_\tau)$ compresses $\bar{\rho}$ below the static spectral radius, the exponential decay rate is accelerated. However, the polynomial prefactor k^{m-1} from the Jordan structure is preserved, making defectiveness the dominant mechanism for extending memory in gated architectures.

E. Why LayerNorm Preserves the Memory Advantage

LayerNorm applied to the hidden state h_t produces $\hat{h}_t = (h_t - \mu_t)\gamma/\sigma_t + \beta$, where μ_t and σ_t are the mean and standard deviation of the components of h_t. This operation:

1. **Preserves direction.** The normalized vector $\hat{h}_t$ retains the relative proportions between components of h_t. In a Jordan block, the "cascade" structure $h_{i,t} = \lambda h_{i,t-1} + h_{i+1,t-1}$ causes signal to flow from higher-index components to lower-index components over time. The relative proportions encode which "stage" of the cascade has been reached, effectively encoding the *age* of the stored memory.
2. **Caps magnitude.** The division by σ_t prevents the polynomial growth k^{m-1} from causing numerical overflow, which would otherwise make training impossible for large m.
3. **Readout compatibility.** The linear readout $y_t = C\hat{h}_t$ can still decode the memory content from the normalized vector, because the information is encoded in the direction (relative proportions) rather than the absolute magnitude.

Without LayerNorm, a 16×16 Jordan block with $\rho = 0.9$ produces hidden states reaching $\sim 10^{26}$ at the peak ($k_{\max} \approx 150$). With LayerNorm, the same structure achieves the best recall accuracy across all sequence lengths.

F. Ablation Study: Block Size and LayerNorm

We provide additional ablation results on the effect of Jordan block size m and LayerNorm.

Table 1. Effect of Jordan block size on associative recall (MSE at $L = 40$, $\rho = 0.9$, averaged over 5 seeds). Without LayerNorm, only small block sizes ($m \leq 4$) are trainable. With LayerNorm, larger blocks consistently improve recall.

Block size m	1	2	3	4	8	16
Without LayerNorm	0.91	0.72	0.54	0.48	NaN	NaN
With LayerNorm	0.91	0.68	0.47	0.41	0.35	0.28

The "sweet spot" of $m = 3$–4 achieves good recall without requiring LayerNorm, providing a practical option when normalization adds undesirable overhead.

G. Experimental Details

G.1. Synthetic Ablations

Implemented in NumPy. We construct exact Jordan blocks of known structure and compute $\|A^k\|_2$ via direct matrix power and `numpy.linalg.norm`. No training or optimization is involved. The three ablation figures validate:

- **Ablation 1** (Figure 1a): The peak location $k_{\max} = (m-1)/(-\ln \rho)$ is confirmed across $\rho \in \{0.8, 0.9, 0.95, 0.99\}$ with $m = 5$.
- **Ablation 2** (Figure 1b): Setting B to have zeros in specific positions reduces the effective block size m', truncating the polynomial degree from k^4 to k^2 to k^0.
- **Ablation 3** (Figure 2): A diagonal matrix with $\lambda = 0.95$ repeated 5 times produces pure exponential decay, while a 5×5 Jordan block with the same eigenvalue produces $k^4\rho^k$ transients. This isolates defectiveness as the causal mechanism.

G.2. PyTorch Experiments

All models use $d_{\text{model}} = 16$, $\rho = 0.9$. Training uses Adam optimizer with learning rate 0.01, gradient clipping at max norm 1.0, and batch size 32. Three initializations are compared:

- **Diagonal**: $A = 0.9 \cdot I_{16}$. Purely diagonal, $m = 1$.
- **HiPPO**: Simplified HiPPO-LegS approximation. The continuous matrix is constructed with $A_{ij} = \sqrt{(2i+1)(2j+1)}$ for $i > j$, $A_{ii} = i+1$, negated, normalized, and discretized as $A_{\text{discrete}} = I + A_{\text{cont}} \cdot (1 - \rho)$.
- **Defective**: $A = 0.9 \cdot I_{16} + N_{16}$, a single 16×16 Jordan block with $\lambda = 0.9$.

Discrete associative recall. Vocabulary size $V = 20$. The task is to recall the value associated with a query key from a sequence of key-value pairs. We use `nn.Embedding`, `CrossEntropyLoss`, 40 epochs per seed, 5 random seeds (seeds 0–4). Sequence lengths $L \in \{10, 20, 30, 40, 50, 60\}$.

Dyck-1 formal language. Binary vocabulary {"(", ")"}. The task is to predict the next token in balanced parenthesis sequences. We use `CrossEntropyLoss`, 40 epochs per seed, 3 random seeds (seeds 0–2). Evaluation metric: exact sequence match accuracy. Sequence lengths $L \in \{10, 15, 20, 25, 30\}$.

G.3. Compute

All experiments run on a single CPU (Intel, 8 GB RAM). Total wall-clock time for the full experimental suite (synthetic ablations + all PyTorch experiments across all seeds) was under 10 minutes.