Robustifying Long-Memory State-Space Models via Hankel Operator Theory

Annan Yu[1], Michael W. Mahoney[2,3,4], N. Benjamin Erichson[2,3]

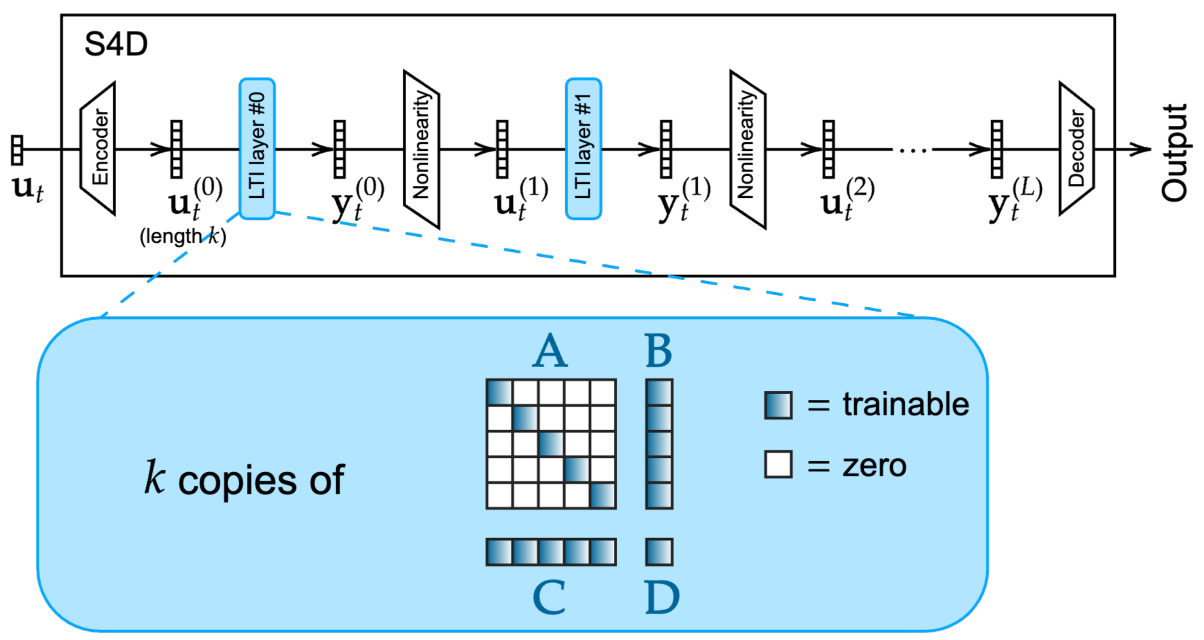
[1]Cornell University [2]Lawrence Berkeley National Laboratory [3]International Computer Science Institute [4]University of California, Berkeley

State-Space Models

State-space models (SSMs) leverage linear, time-invariant (LTI) systems,

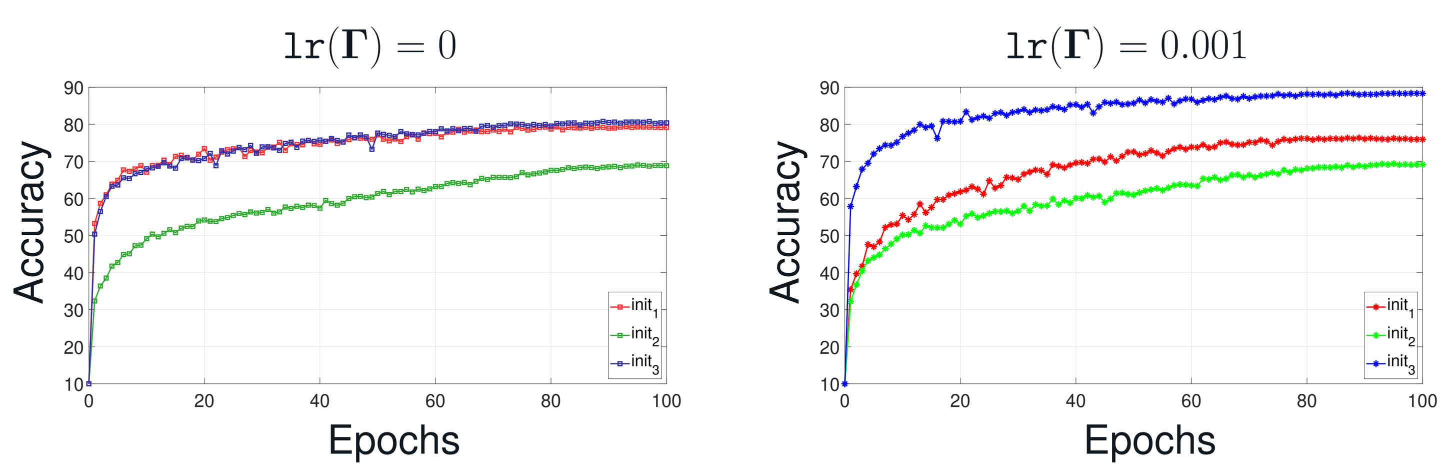
$$\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t),$$

to model long sequential data. In a canonical SSM (e.g. S4D), $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\mathbf{B} \in \mathbb{C}^{n\times 1}$, $\mathbf{C} \in \mathbb{C}^{1\times n}$, and $\mathbf{D} \in \mathbb{C}$ are the trainable parameters.



Initialization and Training Issues

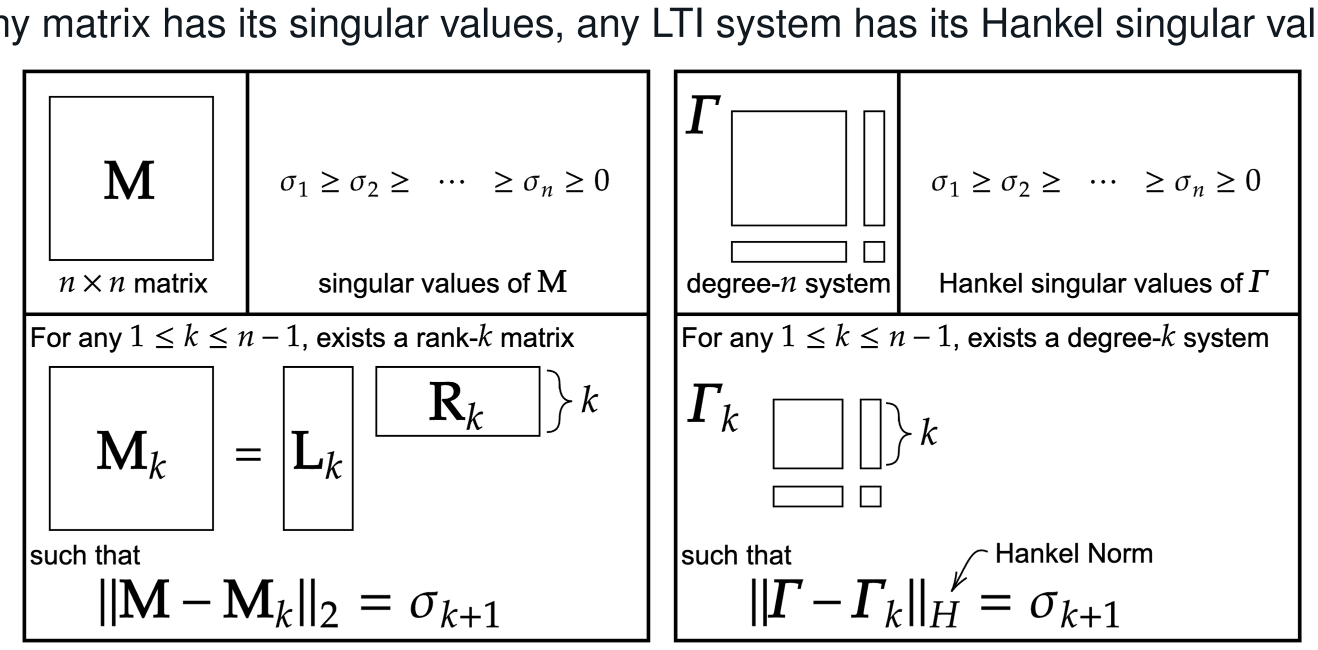
A canonical SSM is highly sensitive to initialization and training hyperparameters. Variations in how the LTI systems are initialized and in the learning rate used to train the system $\Gamma = (\mathbf{A}, \mathbf{B}, \mathbf{C})$ can lead to significantly distinct behaviors on the same task.



In particular, when LTI systems are initialized by $\texttt{init}_1$, $\texttt{init}_2$, and $\texttt{init}_3$, assigning Γ a small learning rate impairs, levels, and improves the performance, respectively.

Towards Better Understanding the Issues: Hankel Singular Values

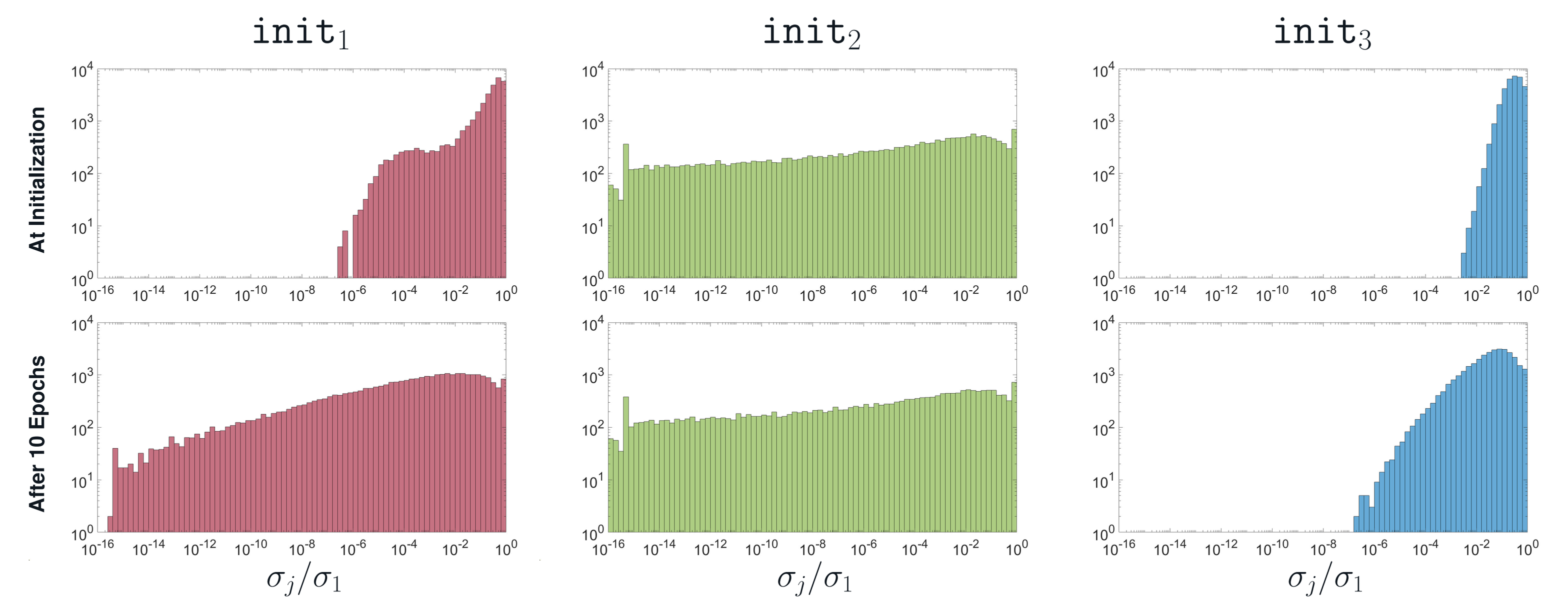
As any matrix has its singular values, any LTI system has its Hankel singular values.



The Hankel singular values tell us "how well we can compress a high-degree LTI system into a low-degree one."

Identify Successful SSMs via Hankel Singular Values

One can use Hankel singular values of LTI systems in an SSM to explain its success or failure. If the Hankel singular values decay fast, then an LTI system is close to a low-degree one, meaning that it has limited expressiveness.



Hankel singular values are large without training the systems Γ. When one starts to train Γ, however, the Hankel singular values decay fast, impairing the performance of the model.

Hankel singular values of the LTI systems always decay fast, making the performance of the SSM suboptimal regardless of whether the systems are trained or not.

Hankel singular values of the systems are large at initialization and remain reasonably high after training. In this case, training the LTI systems accelerates the optimization.

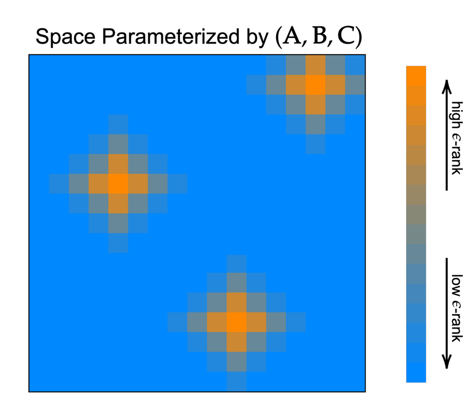
Weakness I: High-Degree LTI Systems are Scarce

From a random matrix theory perspective, one can show that high-degree LTI systems are rare in the parameter space of $(\mathbf{A}, \mathbf{B}, \mathbf{C})$.

The ϵ-rank of a "random" LTI system $\Gamma = (\mathbf{A}, \mathbf{B}, \mathbf{C})$, i.e., the number of Hankel singular values σ_j with

$$\frac{\sigma_j}{\sigma_1} > \epsilon,$$

is roughly $\mathcal{O}(n^{1/2+\text{a bit}})$ with high probability.



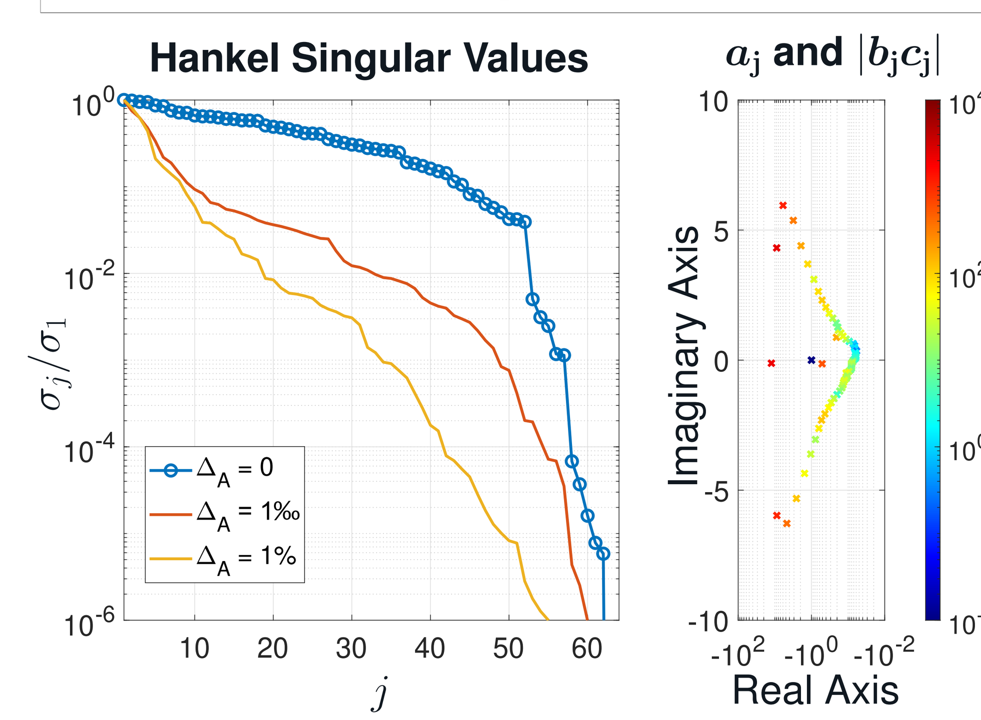
Hence, when training an LTI system parameterized by $(\mathbf{A}, \mathbf{B}, \mathbf{C})$, one is at the risk of losing slow-decaying Hankel singular values.

Weakness II: High-Degree LTI Systems are Numerically Unstable

Suppose we perturb $\mathbf{A} = \text{diag}(a_1, \ldots, a_n)$ by a small amount $\Delta_A > 0$ and $\mathbf{B}$ by $\Delta_B > 0$ to get a perturbed system $\tilde{\Gamma}$. The perturbation of the system is

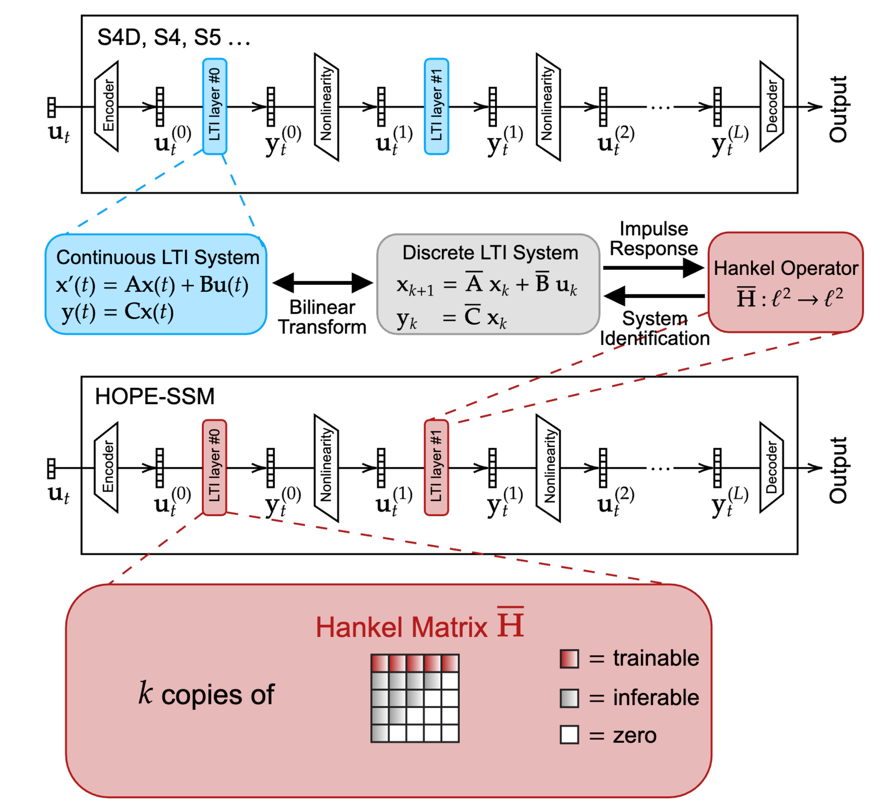
$$\|\Gamma - \tilde{\Gamma}\|_H \leq n\Delta_B \max_j \frac{1}{|\text{Re}(a_j)|} + 4n\Delta_A \max_j \frac{|b_j c_j|}{|\text{Re}(a_j)|^2}.$$

Moreover, this bound is tight up to a factor of n.



Most LTI systems with slow-decaying Hankel singular values have $a_1, \ldots, a_n$ near the imaginary axis; hence, a high-degree system is numerically unstable during training. On the left, one can find the effect of the perturbations on the Hankel singular values of a high-degree system ($\texttt{init}_1$) and the distribution of a_j.

Fix the Weaknesses by Parameterizing with Hankel Operators



Every (discrete) LTI system $\overline{\Gamma} = (\overline{\mathbf{A}}, \overline{\mathbf{B}}, \overline{\mathbf{C}})$ is associated with a doubly infinite Hankel matrix

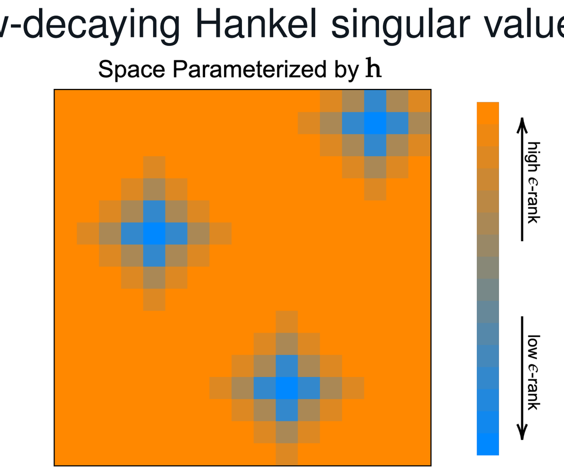
$$\overline{\mathbf{H}}_{ij} = \overline{\mathbf{C}}\overline{\mathbf{A}}^{i+j-2}\overline{\mathbf{B}}.$$

Instead of using $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ to parameterize an LTI system, we propose to use $\mathbf{h} := [h_1 \cdots h_n]^\top$. The only change we introduced is a different way to represent LTI systems by trainable parameters. Importantly, the singular values of the Hankel matrix $\overline{\mathbf{H}}$ are exactly the Hankel singular values of the LTI system. Motivated by the **H**ankel **ope**rator theory, our model is called HOPE.

Benefit I: High-Rank Hankel Operators are Abundant

Just like random matrices, a random Hankel matrix has a high numerical rank with high probability. Hence, one is not at risk of losing slow-decaying Hankel singular values.

Assume $h_1, \ldots, h_n$ are i.i.d. random Gaussian variables. The ϵ-rank of an $n\times n$ random Hankel matrix is almost surely $\Theta(n)$ as $n \to \infty$.



Benefit II: High-Rank Hankel Operators are Numerically Stable

Suppose we perturb $\mathbf{h}$ to $\tilde{\mathbf{h}}$. Let $\mathbf{H}$ be the Hankel matrix defined by $\mathbf{h}$ and $\tilde{\mathbf{H}}$ defined by $\tilde{\mathbf{h}}$. Moreover, let Γ and $\tilde{\Gamma}$ be the corresponding LTI systems. Then, we have

$$\|\Gamma - \tilde{\Gamma}\|_H = \|\mathbf{H} - \tilde{\mathbf{H}}\|_2 \leq \sqrt{n}\|\mathbf{h} - \tilde{\mathbf{h}}\|_2.$$

Benefit III: Hankel Operators Endow SSMs Long-Term Memory

From a continuous-time perspective, the memory of an LTI system parameterized by $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ often has fast (exponentially) decaying memory. The memory of a system parameterized by $\mathbf{h}$ has no decay until $t = n$, after which the system has no memory. Yet, since continuous-time LTI systems in an SSM are discretized with some tunable sampling period $\Delta t > 0$, one can set Δt small to fit the entire sequence into the "memory window" $t \in [0, n]$, even when the sequence length is much larger than n.