Model order reduction of deep structured state-space models: A system-theoretic approach

Marco Forgione, Manas Mejari, and Dario Piga

IDSIA Dalle Molle Institute for Artificial Intelligence USI-SUPSI,
Via la Santa 1, CH-6962 Lugano-Viganello, Switzerland.

March 21, 2024

Abstract

With a specific emphasis on control design objectives, achieving accurate system modeling with limited complexity is crucial in parametric system identification. The recently introduced deep structured state-space models (SSM), which feature linear dynamical blocks as key constituent components, offer high predictive performance. However, the learned representations often suffer from excessively large model orders, which render them unsuitable for control design purposes. The current paper addresses this challenge by means of system-theoretic model order reduction techniques that target the linear dynamical blocks of SSMs. We introduce two regularization terms which can be incorporated into the training loss for improved model order reduction. In particular, we consider modal ℓ_1 and Hankel nuclear norm regularization to promote sparsity, allowing one to retain only the relevant states without sacrificing accuracy. The presented regularizers lead to advantages in terms of parsimonious representations and faster inference resulting from the reduced order models. The effectiveness of the proposed methodology is demonstrated using real-world ground vibration data from an aircraft.

1 Introduction

In recent years, deep structured state-space models (SSM) have emerged as powerful and flexible architectures to tackle machine-learning tasks over sequential data such as time series classification, regression, and forecasting [11, 12, 17, 24]. Notably, they exhibit state-of-the-art performance in problems defined over very long sequences, where Transformers struggle due to their computational complexity that grows quadratically with the sequence length [25].

Essentially, SSMs consist in the sequential connection of linear *dynamical* blocks interleaved with static non-linearities and normalization units, organized in identical repeated layers with skip connections (see Fig. 1). In this sense, they

are closely related to the block-oriented modeling framework [23] traditionally employed by the system identification community, and made compatible for training in a deep-learning setting thanks to the *dynoNet* architecture proposed by some of the authors in [8].

Several mathematical and software implementation solutions have been devised to make learning of SSM architectures–in particular of their key constituent linear dynamical block–fast, well-posed, and efficient. For instance, S4 [12] adopts a continuous-time parameterization, an initialization strategy based on continuous-time memorization theory, and a convolution-based implementation in frequency domain based on fast Fourier transform. Conversely, the deep Linear Recurrent Unit architecture [17] adopts a discrete-time parameterization with a diagonal state-transition matrix, an initialization strategy that constrains the eigenvalues of the system in a region of stability, and an efficient implementation in time domain exploiting the parallel scan algorithm [4].

From the Systems and Control perspective, *parsimonious* representations are often sought, *i.e.*, it is desired to obtain a model that describes system's behaviour with as few parameters and states as possible, to simplify downstream tasks such as controller synthesis, state estimation, etc.

The inadequacy of high-dimensional models has driven growing interest in Model Order Reduction (MOR) techniques. In particular, several contributions focus on reducing the number of states of linear dynamical systems, employing methods that can be broadly categorized into SVD-based and Krylov approximation-based techniques [2]. The former rely on the concept of the *Hankel singular values*, which characterize the complexity of the reduced-order model and provide an error bound of the approximation [9]. These methods include balanced truncation [1, 15], singular perturbation approximation [14] and Hankel norm approximation [18]. On the other hand, Krylov-based approximation methods are iterative in nature. They are based on *moment matching* of the impulse response rather than computation of singular values, see the recent survey paper [21] for an overview of these approaches.

In this paper, we demonstrate the effective adaptation of these MOR techniques, initially designed for linear dynamical systems, to the task of learning simplified deep SSMs architectures while maintaining their predictive capabilities. In principle, an SSM could first be learned using standard machine-learning algorithms, and then each of the constituent linear dynamical blocks could be reduced employing one of the MOR techniques mentioned above. However, we show that the effectiveness of MOR is significantly increased when the low-order modeling objective is already integrated in the training procedure, by means of a *regularization term* in the loss function which promotes parsimonious representations. In particular, we adopt modal ℓ_1 and Hankel nuclear norm regularization approaches that penalize the magnitude of the linear units' eigenvalues and Hankel singular values, respectively. We illustrate our methodology on a well-known system identification benchmark [16] where the aim is to model the oscillations of an aircraft subject to a ground vibration test. We show that specific combinations of regularizers applied during training, along with MOR techniques applied after training, yield the best results.

All our codes are available in the GitHub repository `https://github.com/forgi86/lru-reduction`, allowing full reproducibility of the reported results.

2 Problem Setting

We consider a training dataset consisting of a sequence of input-output samples $\mathcal{D} = \{u_k, y_k\}_{k=1}^N$, generated from an unknown dynamical system $\mathcal{S}$, where $u_k \in \mathbb{R}^{n_u}$ is the input and $y_k \in \mathbb{R}^{n_y}$ is the measured output at time step k. The problem considered in this work is to learn a parametric simulation model $\mathcal{M}(\theta)$ with parameters $\theta \in \Theta$, mapping an input sequence $u_{1:k}$ to the (estimated) output sequence $\hat{y}_{1:k}$, which fits the training dataset $\mathcal{D}$. In particular, the focus is to identify a *parsimonious* model with as few states (in turn, parameters) as possible via regularization and model order reduction techniques.

The parameters θ characterising the model $\mathcal{M}(\theta)$ are estimated according to the criterion:

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \frac{1}{N} \sum_{k=1}^{N} \mathcal{L}\left(y_k, \hat{y}_k(\theta)\right) + \gamma \mathcal{R}(\theta), \tag{1}$$

where $\hat{y}_k(\theta)$ represents the model's output at time k, and $\mathcal{L}(\cdot)$ denotes the chosen fitting loss function. The term $\mathcal{R}(\theta)$ is a regularization cost which aims at enforcing sparsity and reducing the complexity of the model, ultimately facilitating the subsequent MOR step. Different choices for the regularization loss $\mathcal{R}(\theta)$ will be introduced in the Section 4.2.

In this work, $\mathcal{M}(\theta)$ is a SSM architecture recently introduced in [17] and known as deep Linear Recurrent Unit. In the next section, we describe in details the building blocks and parameterizations of this architecture.

3 Deep Structured State-Space Model

The deep Linear Recurrent Unit architecture is visualized in Fig. 1. Its core (shown in gray) is a stack of n_{layers} Linear Recurrent Units (LRU), which are linear dynamical systems, interleaved with static non-linearities, (*e.g.*, Multi Layer Perceptrons (MLP) or Gated Linear Units (GLU) [5]) and normalization units (typically Layer or Batch Normalization [27]), with *skip connections* included in each repeated layer. In particular, the l-th layer of the network with inputs u_k^l and output s_k^l is defined by:

$$\mathcal{M}^l : \begin{cases} \tilde{u}_k^l = \text{Norm}(u_k^l), \\ y_k^l = \text{LRU}(\tilde{u}_{1:k}^l; \theta^l), \\ s_k^l = u_k^l + f(y_k^l), \end{cases} \tag{2}$$

where $\text{Norm}(\cdot)$ is the normalization unit; the LRU is a linear time-invariant (LTI) multi-input multi-output (MIMO) dynamical block whose exact structure

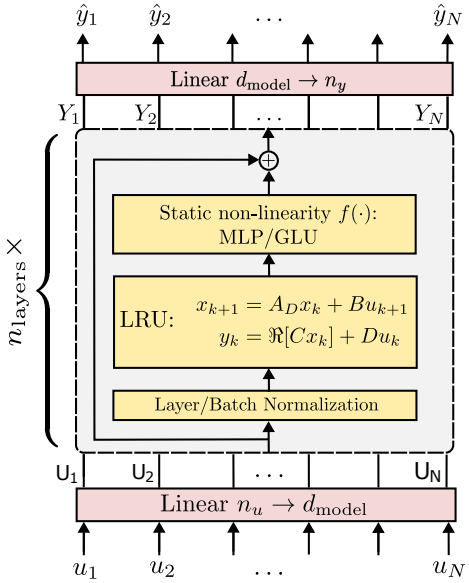

Figure 1: The Deep Linear Recurrent Unit architecture.

is described in the next sub-section; and $f(\cdot) : \mathbb{R}^{d_{\text{model}}} \to \mathbb{R}^{d_{\text{model}}}$ is the static non-linearity, applied in an element-wise fashion to all samples of the LRU output sequence. The first and last transformations in the architecture (pink blocks) are static linear projections mapping the input samples $u_k \in \mathbb{R}^{n_u}$ to $U_k \in \mathbb{R}^{d_{\text{model}}}$, and $Y_k \in \mathbb{R}^{d_{\text{model}}}$ to predicted outputs $\hat{y}_k \in \mathbb{R}^{n_y}$, respectively.

We remark that deep LRU shares a close proximity to the *dynoNet* architecture proposed in [8]. The main difference is that the LRU is a state-space representation of an LTI system, while *dynoNet* employs input-output transfer function descriptions. The architecture is also related to (decoder-only) Transformers [19], with information shared across time steps with an LTI system instead of a causal attention layer.

In the following subsection, we summarize the details of the LRU block. We omit the layer index l when referencing parameters and signals to simplify the notation. Furthermore, we redefine u_k/y_k as the input/output samples of the LRU to match the standard notation of linear system theory.

Linear Recurrent Unit

The LRU is a linear, discrete-time, MIMO dynamical system described in state-space form as:

$$x_k = A_D \; x_{k-1} + B u_k, \tag{3a}$$
$$y_k = \Re[C x_k] + D u_k, \tag{3b}$$

where $\Re[\cdot]$ denotes the real part of its argument, $A_D \in \mathbb{C}^{n_x \times n_x}$, $B \in \mathbb{C}^{n_x \times n_u}$ and $C \in \mathbb{C}^{n_y \times n_x}$ are complex-valued matrices, and $D \in \mathbb{R}^{n_y \times n_u}$ is a real-valued matrix. The matrix A_D has a diagonal structure:

$$A_D = \text{diag}(\lambda_1, \dots, \lambda_{n_x}), \tag{4}$$

where $\lambda_j,\ j = 1, \dots, n_x$ are the complex eigenvalues, or modes, of the system, which is thus represented in a *modal* form. In order to guarantee asymptotic stability of the system, *i.e*, to enforce $|\lambda_j| < 1,\ j = 1, \dots, n_x$, each eigenvalue $\lambda_j \in \mathbb{C}$ is, in turn, parameterized as $\lambda_j = \exp(-\exp(\nu_j) + i \exp(\phi_j))$, where $\nu_j > 0$. Note that since $\exp(\nu_j) > 0$ for $\nu_j > 0$, this ensures $|\lambda_j| = \exp(-\exp(\nu_j)) < 1$.

The input matrix B is parameterized as $B = \text{diag}(\gamma_1, \dots, \gamma_{n_x})\tilde{B}$, where $\gamma_j = \sqrt{1 - |\lambda_j|^2},\ j = 1, \dots, n_x$, is a normalization factor introduced to obtain state signals with the same power as that of the input signal. Overall, the learnable parameters of a single LRU block are $\theta := \{\{\nu_j, \phi_j\}_{j=1}^{n_x}, \tilde{B}, C, D\}$.

Remark 1 *System* (3) *has an equivalent complex-conjugate representation:*

$$\tilde{x}_k = \begin{bmatrix} A_D & 0 \\ 0 & A_D^* \end{bmatrix} \tilde{x}_{k-1} + \begin{bmatrix} B \\ B^* \end{bmatrix} u_k, \tag{5a}$$

$$y_k = \frac{1}{2} \begin{bmatrix} C & C^* \end{bmatrix} \tilde{x}_k + D u_k, \tag{5b}$$

with $\tilde{x}_k \in \mathbb{C}^{2n_x}$, which in turn may be transformed in a real Jordan form, with a block-diagonal state-transition matrix containing n_x 2x2 blocks, see e.g. Appendix E.3 of [17]. The complex-valued, diagonal representation (3) *is preferred for its implementation simplicity and halved state dimension.*

4 Model Order Reduction and Regularization

In this section, we provide a brief review of the MOR techniques used in this paper. Next, we introduce regularization techniques aimed at promoting the learning of parsimonious LRU representations in terms of state complexity.

4.1 Reduction by truncation and singular perturbation

Order reduction techniques based on truncation decrease the dimensionality of a dynamical system by eliminating states that, for various reasons, are considered less important [10]. Consider a state-space model G with realization:

$$\begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_{1,k-1} \\ x_{2,k-1} \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u_k, \tag{6a}$$

$$y_k = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + D u_k, \tag{6b}$$

where the partitioning corresponds to important states to be kept $x_1 \in \mathbb{R}^r$ and unimportant states to be removed $x_2 \in \mathbb{R}^{n_x - r}$, respectively. The state-space *truncation* method approximates (6) with a reduced-order system G_r having state-space matrices:

$$A_r = A_{11}, B_r = B_1, C_r = C_1, D_r = D. \tag{7}$$

Note that state-space truncation does not preserve the steady-state behavior of G and, in general, it may alter its low-frequency response significantly. Alternatively, the removed states x_2 may be set to equilibrium by solving for $x_{2,k} = x_{2,k-1}$ in (6a). This results in the so-called *singular perturbation* approximation, where the reduced-order system G_r is defined by the following state-space matrices:

$$A_r = A_{11} + A_{12}(I - A_{22})^{-1}A_{21} \tag{8a}$$

$$B_r = B_1 + A_{12}(I - A_{22})^{-1}B_2 \tag{8b}$$

$$C_r = C_1 + C_2(I - A_{22})^{-1}A_{21} \tag{8c}$$

$$D_r = C_2(I - A_{22})^{-1}B_2 + D. \tag{8d}$$

Singular perturbation preserves the steady-state behavior of (6), and generally provides a better match in the lower frequency range with respect to plain state-space truncation.

In the control literature, state-space truncation and singular perturbation approximations are typically applied to systems described either in modal or in *balanced* realization form [10]. The resulting MOR techniques will be denoted here as modal truncation (MT), modal singular perturbation (MSP), balanced truncation (BT), and balanced singular perturbation (BSP). In the modal methods, the states are sorted according to non-increasing magnitude of the corresponding eigenvalues, so that the fastest dynamics can be discarded. This choice is often motivated by physical arguments, when the fast dynamics are associated to uninteresting second-order effects (e.g., electrical dynamics in mechanical systems). In balanced methods, the states are sorted for non-increasing value of the corresponding Hankel singular values. This choice is supported by the following approximation bound, which holds both for BT and BSP:

$$\|G - G_r\|_\infty \leq 2 \sum_{j=r+1}^{n_x} \sigma_j, \tag{9}$$

where $\|\cdot\|_\infty$ denotes the $\mathcal{H}_\infty$ norm and $\sigma_j,\ j = r+1, \ldots, n_x$ are the Hankel singular values corresponding to the eliminated states, see [13, Lemma 3.7].

For the LRU, MT/MSP are directly applicable, being the block already represented in a modal form. Conversely, BT/BSP require a three-step procedure where (i) system (3) is first transformed to a (non-diagonal) balanced form, then (ii) reduced according to either (7) or (8), and finally (iii) re-diagonalized with a state transformation obtained from the eigenvalue decomposition of the state matrix of the system obtained at step (ii) to fit the LRU famework.

4.2 Regularized Linear Recurrent Units

In this section, we introduce two regularization approaches that promote learning of LRUs with a reduced state complexity. These methods leverage system-theoretic MOR methods described in the previous sub-section and exploit the diagonal structure of the LRU's state-transition matrix A_D.

4.2.1 Modal ℓ_1-regularization

As mentioned in Section 4.1, high-frequency modes often correspond to secondary phenomena that may be eliminated without compromising the salient aspects of the modeled dynamics. For discrete-time LTI systems, fast dynamics are associated to eigenvalues whose modulus $|\lambda_j|$ is small. An ℓ_1-regularization is therefore introduced to push some of the modes towards zero during training:

$$\mathcal{R}(\theta) = \sum_{l=1}^{n_{\text{layers}}} \sum_{j=1}^{n_x} |\lambda_j^l|. \tag{10}$$

Indeed, ℓ_1-regularization is known to promote sparsity in the solution [26]. The states corresponding to eigenvalues that are closer to zero will then be eliminated with a MOR method at a second stage after training. Note that modal ℓ_1-regularization of the LRU is computationally cheap, as the eigenvalues are directly available on the diagonal of the state-transition matrix A_D.

4.2.2 Hankel nuclear norm regularization

It is known that the McMillan degree (minimum realization order) of a discrete-time LTI system coincides with the rank of its associated (infinte-dimensional) Hankel operator H [13]. The (i, j)-th block of H is defined as $H_{ij} = g_{i+j-1}$, where $g_k = CA_D^{k-1}B$ is the impulse response coefficient at time step k. Minimizing the rank of the Hankel operator thus aligns with the objective of obtaining a low-order representation. However, the rank minimization problem is hard to solve and the *nuclear norm* of the Hankel operator $\|H(g)\|_* := \sum_j \sigma_j$, defined as the sum of its singular values σ_j, is often used as a *convex surrogate* of the rank [6, 18].

Following this rationale, we employ the Hankel nuclear norm of the LRUs as a regularization term in training:

$$\mathcal{R}(\theta) = \sum_{l=1}^{n_{\text{layers}}} \sum_{j=1}^{n_x} \sigma_j^l, \tag{11}$$

where σ_j^l denotes the j-th singular value of the Hankel operator of the LRU in the l-th layer. Note that, as $\sigma_j^l \geq 0,\ j = 1, \ldots, n_x$, the term $\sum_{j=1}^{n_x} \sigma_j^l$ in (11) is the ℓ_1-norm of the Hankel singular values, thus, promoting sparsity.

It can be proved that the j-th singular value of the Hankel operator is given by $\sigma_j(H) = \sqrt{\texttt{eig}_j(PQ)}$, where P and Q are the controllability and observability Grammians of the LTI model [13]. In appendix A.1, we show how the

Grammians P and Q, and in turn the Hankel singular values can be computed efficiently for systems in modal form.

Remark 2 (ℓ_2-regularization) *If the ℓ_2-norm of the Hankel singular values is considered in* (11) *instead of the ℓ_1-norm, the computational burden during training can be further reduced exploiting the identity $\sum_{j=1}^{n_x} \sigma_j^2 = \sum_{j=1}^{n_x} \texttt{eig}_j(PQ) = \text{trace}(PQ)$. Thus, differentiation of the eigenvalues of PQ is not required. Nonetheless, it is known that ℓ_2-norm regularization does not enforce sparsity in the solution, contrary to the ℓ_1 case.*

Remark 3 ($\mathcal{H}_\infty$-error bound) *The use of the regularizer* (11) *is further motivated by the $\mathcal{H}_\infty$ error bound* (9)*. This suggests to combine Hankel-norm regularization during training with MOR based on either BT or BSP.*

5 Case Study

We test the methodologies of the paper on the ground vibration dataset of an F-16 aircraft introduced in [16].

The input $u \in \mathbb{R}$ (N) is the force generated by a shaker mounted on the aircraft's right wing, while the outputs $y \in \mathbb{R}^3$ (m/s^2) are the accelerations measured at three test locations on the aircraft. Input and output samples are collected at a constant frequency of 400 Hz. We train deep LRU models with structure as shown in Fig. 1 characterized by: Layer Normalization; MLP non-linearity; $d_{\text{model}} = 50$; $n_x = 100$; and $n_{\text{layers}} = 6$. The MLP has one hidden layer with 400 hidden units and Gaussian Error Linear Unit (GELU) non-linearities.

Training is repeated three times with identical settings except for the regularization strategy, which is set to: (i) no regularization, (ii) modal ℓ_1-regularization, and (iii) Hankel nuclear norm regularization. For both (ii) and (iii), the regularization strength is set to $\gamma = 10^{-2}$. We train the models on all the input/output sequences suggested for training in [16] except the one denoted as "Special Odds". To take advantage of parallel computations on more sequences, we split the datasets in (partially overlapping) sub-sequences of length $N = 5000$ samples each and compute the training loss (1) over batches of 64 sub-sequences simultaneously. In the definition of the loss, the first 200 samples of each sub-sequence are discarded to cope with the state initialization issue, according to the ZERO initialization scheme described in [7]. We minimize the mean squared error training loss over 10 epochs of AdamW with constant learning rate 10^{-4}, where at each epoch all the 688820 sub-sequences of length N in the training data are processed.

We report the *fit* index [22], the Root Mean Squared Error (RMSE), and the Normalized Root Mean Squared Error (NRMSE) on the three output channels in Table 1. For brevity, we exclusively report the results obtained on the test dataset denoted as "FullMSine_Level6_Validation". The three trained models achieve similar performance, which is also in line with existing state-of-the-art. For instance, the average NRMSE over the three channels is about 0.15, which

Regularization	Channel 1			Channel 2			Channel 3		
	fit	RMSE	NRMSE	*fit*	RMSE	NRMSE	*fit*	RMSE	NRMSE
No reg.	86.5	0.180	0.134	90.0	0.167	0.099	76.2	0.368	0.237
Modal ℓ_1	85.4	0.195	0.145	89.8	0.171	0.102	74.5	0.395	0.254
Hankel norm	85.8	0.190	0.142	89.0	0.185	0.110	75.5	0.379	0.245

Table 1: Performance of the SSM trained with different regularization methods.

is close to the result reported in [20]. However, we observe that regularization has a strong effect on the properties of the estimated LTI blocks.

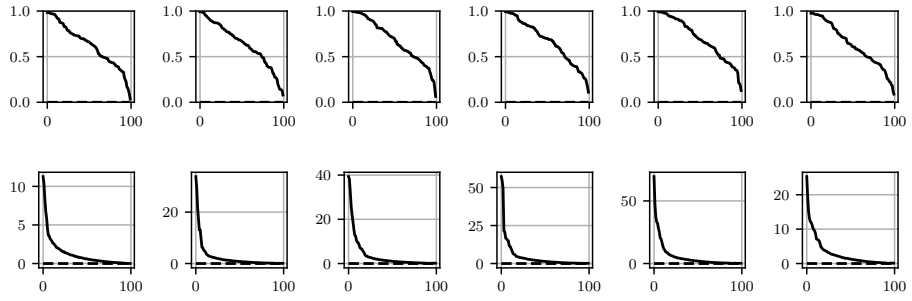

Figure 2: No regularization: eigenvalues magnitude (top) and Hankel singular values (bottom).

The plots in the six columns of Fig. 2, 3 and 4 illustrate this effect, where each column corresponds to LRU in one of the 6 layers. For the model without regularization (Fig. 2), most of the eigenvalues have non-zero magnitude (top panel). In this sense, the modal reduction methods MT/MSP are not expected to be effective. The Hankel singular values decrease slightly faster towards zero, suggesting that the effectiveness of the balanced reduction methods BT/BSP might be marginally superior. As for the model obtained with modal ℓ_1-regularization (Fig. 3), several eigenvalues have been pushed towards zero (top panel), suggesting the potential effectiveness of modal reduction methods. Finally, for the model trained with Hankel nucelar norm regularization (Fig. 4), the Hankel singular values decrease very sharply towards zero (bottom panel), while none of the eigenvalues' magnitude is pushed towards zero. Thus, we expect balanced reduction methods to be effective in this case.

In Table 2, we report the maximum number of eliminated states with the different MOR techniques applied to the three trained models, such that the performance degradation in test (in terms of *fit* index) averaged over the three output channels is less than 1%. The best results are obtained for the combinations of modal ℓ_1-regularization followed by MSP and Hankel nuclear norm regularization followed by BSP, which both lead to 91 eliminated states. We

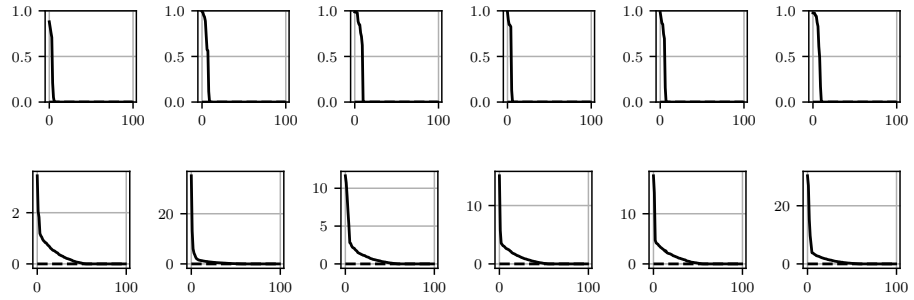

Figure 3: Modal ℓ_1 regularization: eigenvalues magnitude (top) and Hankel singular values (bottom).

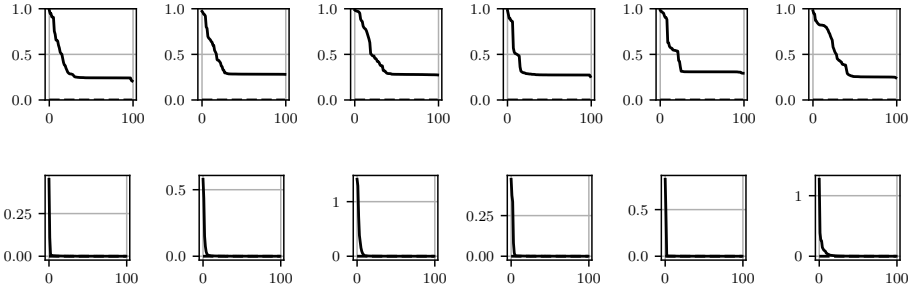

Figure 4: Hankel nuclear norm regularization: eigenvalues magnitude (top) and Hankel singular values (bottom).

also observe that, when regularization is not applied in training, the subsequent MOR is decisively less effective. Fig. 5 further highlights this key experimental results: when training with no regularization, the best reduction approach (BSP) is significantly less effective than the optimal regularizer+MOR combinations: modal ℓ_1+MSP and Hankel nuclear norm+BSP.

	Truncation Method			
Regularization Method	BT	BSP	MT	MSP
No Regularization	28	43	3	35
Modal ℓ_1	56	73	0	**91**
Hankel nuclear norm	89	**91**	18	76

Table 2: Maximum number of modes that can be eliminated while keeping the performance of the trained model within 1% of the full case.

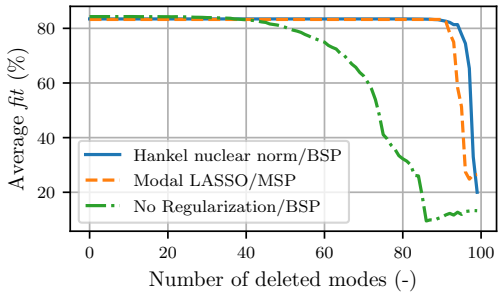

Figure 5: Test performance obtained for increasing number of removed modes in all layers, for selected combinations of regularization and model order reduction approaches.

6 Conclusions

We have presented regularization methods and model order reduction approaches that enable substantial simplification of deep structured state-space models. Our experimental results suggest that regularization is a fundamental ingredient of our procedure. Indeed, model order reduction executed as a mere post-hoc step, after a standard training conducted without regularization appears to be significantly less effective. In future works, we will analyze in more depth the effect of the regularization strength γ through more extensive numerical experiments and possibly with analytical tools. Moreover, we aim at decreasing the number of internal input/output channels d_{model} and of layers n_{layers} of the architecture. A possible approach is based on group LASSO regularization, following an approach similar to the one recently introduced in [3] for more classical neural state-space models. Finally, we will extend our techniques to other architectures that feature linear dynamical blocks at their core, such as dynoNet and deep Koopman representations.

A Appendix

A.1 Computation of Hankel singular values

The Hankel singular values of a discrete-time LTI system with complex-valued matrices (A, B, C, D) are given by:

$$\sigma_j = \sqrt{\texttt{eig}_j(PQ)}, \quad \forall j \in 1, \dots, n_x, \tag{12}$$

where $P \in \mathbb{C}^{n_x \times n_x}$ and $Q \in \mathbb{C}^{n_x \times n_x}$ are the controllability and observability Grammians, respectively, which are the solutions of the discrete Lyapunov equations [13]:

$$APA^* - P + BB^* = 0 \tag{13}$$

$$A^*QA - Q + C^*C = 0, \tag{14}$$

A.2 Solution to a diagonal Discrete Lyapunov equation

We show that discrete Lyapunov equations can be solved efficiently for systems in modal representation where matrix A is diagonal. The direct method to solve the Lyapunov equation with variable X:

$$AXA^* - X + Y = 0 \tag{15}$$

is obtained by exploiting the product property:

$$\text{vec}(AXB) = (B^\top \otimes A)\text{vec}(X), \tag{16}$$

where $\otimes$ is the Kronecker product operator and $\text{vec}(\cdot)$ represents the *column-wise* vectorization operation. Applying this formula to (15), one obtains:

$$(I - A^* \otimes A)\text{vec}(X) = \text{vec}(Y), \tag{17}$$

which is a linear system in the unknowns $\text{vec}(X)$. If A is diagonal, the left-hand side matrix of (17) is also diagonal, and thus its solution is simply obtained through n_x^2 scalar divisions.

References

[1] U.M. Al-Saggaf and G. F. Franklin. Model reduction via balanced realizations: an extension and frequency weighting techniques. *IEEE Transactions on Automatic Control*, 33(7):687–692, 1988.

[2] A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics, 2005.

[3] A. Bemporad. Linear and nonlinear system identification under ℓ_1- and group-Lasso regularization via L-BFGS-B. *arXiv preprint arXiv:2403.03827*, 2024.

[4] G. E. Blelloch. Prefix sums and their applications. 1990.

[5] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In *International Conference on Machine Learning*, pages 933–941. PMLR, 2017.

[6] M. Fazel, H. Hindi, and S.P. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *Proc. of the American Control Conf.*, volume 6, pages 4734–4739, 2001.

[7] M. Forgione, M. Mejari, and D. Piga. Learning neural state-space models: do we need a state estimator? *arXiv preprint arXiv:2206.12928*, 2022.

[8] M. Forgione and D. Piga. *dynoNet*: A neural network architecture for learning dynamical systems. *International Journal of Adaptive Control and Signal Processing*, 35(4):612–626, 2021.

[9] K. Glover. All optimal hankel-norm approximations of linear multivariable systems and their L^∞-error bounds. *International Journal of Control*, 39(6):1115–1193, 1984.

[10] M. Green and D. Limebeer. *Linear Robust Control*. Dover publications, 2012.

[11] A. Gu, K. Goel, A. Gupta, and C. Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022.

[12] A. Gu, K. Goel, and C. Ré. Efficiently modeling long sequences with structured state spaces. *The International Conference on Learning Representations (ICLR)*, 2022.

[13] T. Katayama. *Subspace Methods for System Identification*. Springer London, 2005.

[14] Y. Liu and B. O. D. Anderson. Singular perturbation approximation of balanced systems. *International Journal of Control*, 50(4):1379–1405, 1989.

[15] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17–32, 1981.

[16] J. P. Noël and M. Schoukens. F-16 aircraft benchmark based on ground vibration test data. In *2017 Workshop on Nonlinear System Identification Benchmarks*, pages 19–23, 2017.

[17] A. Orvieto, S. L. Smith, A. Gu, A. Fernando, C. Gulcehre, R. Pascanu, and S. De. Resurrecting recurrent neural networks for long sequences. *arXiv preprint arXiv:2303.06349*, 2023.

[18] G. Pillonetto, T. Chen, A. Chiuso, G. De Nicolao, and L. Ljung. Regularized linear system identification using atomic, nuclear and kernel-based norms: The role of the stability constraint. *Automatica*, 69:137–149, 2016.

[19] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[20] M. Revay, R. Wang, and I. R. Manchester. Recurrent equilibrium networks: Flexible dynamic models with guaranteed stability and robustness. *IEEE Transactions on Automatic Control*, 2023.

[21] G. Scarciotti and A. Astolfi. Interconnection-based model order reduction - a survey. *European Journal of Control*, 75:100929, 2024.

[22] J. Schoukens and L. Ljung. Nonlinear system identification: A user-oriented road map. *IEEE Control Systems Magazine*, 39(6):28–99, 2019.

[23] M. Schoukens and K. Tiels. Identification of block-oriented nonlinear systems starting from linear approximations: A survey. *Automatica*, 85:272–292, 2017.

[24] J. TH. Smith, A. Warrington, and S. W. Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022.

[25] Y. Tay, M. Dehghani, S. Abnar, D. Shen, Y.and Bahri, P. Pham, J. Rao, L. Yang, S. Ruder, and D. Metzler. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*, 2020.

[26] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288, 1996.

[27] Y. Wu and K. He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.