Towards context-aware learning for control: Balancing stability and model-learning error

Nitin Shyamkumar* Serkan Gugercin† Benjamin Peherstorfer*

March 3, 2022

Abstract

Classical data-driven control typically follows the learn-then-stabilize scheme where first a model of the system of interest is identified from data and then a controller is constructed based on the learned model. However, learning a model from data is challenging since it can incur high training costs and the model quality critically depends on the available data. In this work, we address how well one needs to learn a model to derive a controller by formalizing the trade off between learning error and controller performance in the specific setting of robust $\mathcal{H}_\infty$ control. We propose a bound on the stability radius of a robust controller with respect to the error of the learned model. The proposed analysis suggests that tolerating an increased learning error leads to a small decrease in the performance objective of the controller. Numerical experiments with systems from aerospace engineering demonstrate that judiciously balancing learning error and control performance can indeed reduce the number of data points by one order of magnitude with less than 5% decrease in control performance as measured with the $\mathcal{H}_\infty$ stability radius.

1 Introduction

With a deluge of data and a lack of models, learning controllers from data is becoming an ever more important challenge in computational science and engineering. Typically, data-driven control consists of two steps: First, a model of the system of interest is learned from data, which is a process referred to as system identification. Second, a controller is constructed based on the learned model. The learned controller is then applied to the system of interest.

Intuitively, one expects that the quality of the controller depends on how well the learned model approximates the system dynamics. In this work, we formalize this intuition: We consider $\mathcal{H}_\infty$ control and measure the quality of a learned controller by its stability radius when applied to the system. We present an analysis that bounds the stability radius with respect to the error of the learned model. Our bound gives rise to a trade off between stability radius and model error that allows relaxing model accuracy in favor of a lower stability radius (poorer control performance); this might be beneficial in cases where learning a model is challenging in terms of data requirements and training costs. Understanding how model-learning errors influence controller quality is an important step towards context-aware learning for control, where models are learned explicitly for deriving controllers. Context-aware learning has previously been developed for uncertainty quantification [19, 1].

Our contributions are: **(1)** We prove that under certain conditions, the model-learning error can be compensated by specific disturbance signals. **(2)** We establish a trade off between model-learning error and stability radius in $\mathcal{H}_\infty$ control. **(3)** We balance model-learning error and stability radius when learning a model with least-squares regression and show that in our numerical example tolerating a small decrease in the stability radius can lead to one order of magnitude reduction in the required number of data points.

Literature review: There is a large body of literature on constructing reduced models for control; see the surveys [6, 23, 7, 13]. However, classical model reduction methods motivated by control, such as balanced truncation, are intrusive in the sense that a model of the system must be available. Only recently have works started exploring novel reformu-

*Courant Institute of Mathematical Sciences, New York University

†Department of Mathematics and the Division of Computational Modeling and Data Analytics, Virginia Tech

lations of balanced truncation to learn models from data [12]. Other non-intrusive model reduction methods are, e.g., the Loewner framework [2, 17, 5, 11], sparse methods [8], dynamic mode decomposition and operator inference [24, 26, 20], lift & learn [21], and methods based on regressing reduced coefficients [4, 15, 25]. However, they are all generic data-driven modeling techniques independent of the control task. There are also works on reducing controllers after they have been derived [28] and theoretical studies on the control performance of interconnected systems with reduced components [22]. None of these works explicitly studies the trade off between model-learning error and controller quality. From the machine learning community, there have been works on balancing training and performance metrics such as the error in the $\mathcal{H}_\infty$ norm for finite-impulse-response filter models for single-input-single-output systems [27]. In contrast, our results are applicable to broader data-driven modeling methods. In [9], the authors establish an end-to-end linear-quadratic-controller framework with probabilistic error bounds, whereas we focus on robust control in $\mathcal{H}_\infty$.

2 Preliminaries and problem formulation

2.1 True and learned system

Consider a linear time-invariant (LTI) dynamical system Σ with control input $\boldsymbol{u}(t) \in \mathbb{R}^{n_u}$ and control output $\boldsymbol{y}(t) \in \mathbb{R}^{n_y}$. The system Σ also has a disturbance input $\boldsymbol{w}(t) \in \mathbb{R}^{n_w}$ and a disturbance output $\boldsymbol{z}(t) \in \mathbb{R}^{n_z}$. We want to control the system Σ via the control input $\boldsymbol{u}$ and control output $\boldsymbol{y}$; however, a model for Σ is unavailable. Therefore, we learn a system $\widetilde{\Sigma}$ using data collected from the system Σ. A state-space model of this learned system $\widetilde{\Sigma}$ is

$$\widetilde{\Sigma} = \left\{ \begin{bmatrix} \dot{\tilde{\boldsymbol{x}}}(t) \\ \boldsymbol{z}(t) \\ \boldsymbol{y}(t) \end{bmatrix} = \left[\begin{array}{c|cc} \widetilde{\boldsymbol{A}} & \widetilde{\boldsymbol{B}}_1 & \widetilde{\boldsymbol{B}} \\ \hline \widetilde{\boldsymbol{C}}_1 & \mathbf{0} & \mathbf{0} \\ \widetilde{\boldsymbol{C}} & \mathbf{0} & \mathbf{0} \end{array}\right] \begin{bmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{w}(t) \\ \boldsymbol{u}(t) \end{bmatrix} \right. , \quad (1)$$

with state $\tilde{\boldsymbol{x}}(t) \in \mathbb{R}^{n_x}$.

From the learned system $\widetilde{\Sigma}$, we derive a controller

$$\widetilde{K} = \left\{ \begin{bmatrix} \dot{\tilde{\boldsymbol{x}}}_c(t) \\ \boldsymbol{u}(t) \end{bmatrix} = \left[\begin{array}{c|c} \widetilde{\boldsymbol{K}}_A & \widetilde{\boldsymbol{K}}_B \\ \hline \widetilde{\boldsymbol{K}}_C & \widetilde{\boldsymbol{K}}_D \end{array}\right] \begin{bmatrix} \tilde{\boldsymbol{x}}_c(t) \\ \boldsymbol{y}(t) \end{bmatrix} \right. , \quad (2)$$

with state $\tilde{\boldsymbol{x}}_c(t)$ of dimension n_c. The special case of a zero order controller consists of a static output feedback control system described by $\boldsymbol{u}(t) = \widetilde{\boldsymbol{K}}_D \boldsymbol{y}(t)$ ($n_c = 0$). The learned system $\widetilde{\Sigma}$ connected to the controller $\widetilde{K}$ results in the controlled learned system

$$\widetilde{\Sigma}_{\widetilde{K}} = \left\{ \begin{bmatrix} \dot{\overline{\boldsymbol{x}}}(t) \\ \boldsymbol{z}(t) \end{bmatrix} = \left[\begin{array}{c|c} \widetilde{\boldsymbol{A}}_{\widetilde{K}} & \widetilde{\boldsymbol{B}}_{\widetilde{K}} \\ \hline \widetilde{\boldsymbol{C}}_{\widetilde{K}} & \mathbf{0} \end{array}\right] \begin{bmatrix} \overline{\boldsymbol{x}}(t) \\ \boldsymbol{w}(t) \end{bmatrix} \right. , \quad (3)$$

with

$$\widetilde{\boldsymbol{A}}_{\widetilde{K}} = \begin{bmatrix} \widetilde{\boldsymbol{A}} + \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{K}}_D\widetilde{\boldsymbol{C}} & \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{K}}_C \\ \widetilde{\boldsymbol{K}}_B\widetilde{\boldsymbol{C}} & \widetilde{\boldsymbol{K}}_A \end{bmatrix} , \quad (4)$$

$$\widetilde{\boldsymbol{B}}_{\widetilde{K}} = \begin{bmatrix} \widetilde{\boldsymbol{B}}_1 \\ \mathbf{0} \end{bmatrix}, \text{ and } \widetilde{\boldsymbol{C}}_{\widetilde{K}} = \begin{bmatrix} \widetilde{\boldsymbol{C}}_1 & \mathbf{0} \end{bmatrix} . \quad (5)$$

With linear feedback $\boldsymbol{w}(t) = \widetilde{\boldsymbol{\Delta}}\boldsymbol{z}(t)$ between the disturbance input $\boldsymbol{w}(t)$ and the disturbance output $\boldsymbol{z}(t)$, we obtain the closed-loop learned system $\dot{\overline{\boldsymbol{x}}}(t) = \widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}})\overline{\boldsymbol{x}}(t)$ with system matrix

$$\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) = \widetilde{\boldsymbol{A}}_{\widetilde{K}} + \widetilde{\boldsymbol{B}}_{\widetilde{K}}\widetilde{\boldsymbol{\Delta}}\widetilde{\boldsymbol{C}}_{\widetilde{K}} . \quad (6)$$

The stability radius of the closed-loop learned system is

$$\widetilde{\gamma} = \sup_{\gamma} \left\{ \gamma : \|\boldsymbol{\Delta}\|_2 < \gamma \text{ with } \alpha\left(\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\boldsymbol{\Delta})\right) < 0 \right\} , \quad (7)$$

where $\alpha(\cdot)$ is the spectral abscissa, which is the largest of the real parts of the eigenvalues of the matrix argument.

2.2 Robust $\mathcal{H}_\infty$ control

Let $\widetilde{H}_{\widetilde{K}}$ be the transfer function of the controlled learned system $\widetilde{\Sigma}_{\widetilde{K}}$. An $\mathcal{H}_\infty$-optimal controller minimizes the objective $\widetilde{K} \in \inf_K \|\widetilde{H}_K\|_{\mathcal{H}_\infty}$, with norm

$$\left\|\widetilde{H}_K\right\|_{\mathcal{H}_\infty} = \begin{cases} \sup_{\omega\in\mathbb{R}} \sigma_{\max}(\widetilde{H}_K(\mathbf{i}\omega)) , & \text{if } \alpha\left(\widetilde{\boldsymbol{A}}_K\right) < 0 , \\ \infty , & \text{else} , \end{cases}$$

where the imaginary unit is $\mathbf{i} = \sqrt{-1}$ and $\sigma_{\max}(\cdot)$ is the largest singular value of the argument.

2.3 Problem Formulation

Classical learn-then-stabilize approaches learn $\widetilde{\Sigma}$ and then compute $\widetilde{K}$ using $\widetilde{\Sigma}$. The learned controller $\widetilde{K}$ is then applied to the true system Σ, resulting in the controlled true system $\Sigma_{\widetilde{K}}$. We explore two questions about such learn-then-stabilize approaches: (1) How can we trade learning error in $\widetilde{\Sigma}$ for stability when our stated goal is a stable true system $\Sigma_{\widetilde{K}}$? (2) What system properties characterize how easy it is to trade stability for learning error?

3 Bounding stability of controlled system

3.1 Equivalence of learning error and feedback disturbance

We start our analysis by modeling the error of the state-space model (1) of the learned system as additive perturbations. Let

$$\Sigma = \left\{ \begin{bmatrix} \dot{\boldsymbol{x}}(t) \\ \boldsymbol{z}(t) \\ \boldsymbol{y}(t) \end{bmatrix} = \left[\begin{array}{c|cc} \boldsymbol{A} & \boldsymbol{B}_1 & \boldsymbol{B} \\ \hline \boldsymbol{C}_1 & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} & \boldsymbol{0} \end{array}\right] \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{w}(t) \\ \boldsymbol{u}(t) \end{bmatrix} \right. ,$$

be an—in practice unknown—state-space model of the true system with state $\boldsymbol{x}(t) \in \mathbb{R}^{n_x}$ that is realized in the same coordinate system as the learned model. We assume that the state space model corresponds to the minimal realization of the true system. The state-space model of the true system Σ and the learned system $\widetilde{\Sigma}$ are related as $\widetilde{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{\delta}_A$, $\widetilde{\boldsymbol{B}} = \boldsymbol{B} + \boldsymbol{\delta}_B$, $\widetilde{\boldsymbol{C}} = \boldsymbol{C} + \boldsymbol{\delta}_C$, $\widetilde{\boldsymbol{B}}_1 = \boldsymbol{B}_1 + \boldsymbol{\delta}_{B_1}$, and $\widetilde{\boldsymbol{C}}_1 = \boldsymbol{C}_1 + \boldsymbol{\delta}_{C_1}$, where $\boldsymbol{\delta}_A$, $\boldsymbol{\delta}_B$, $\boldsymbol{\delta}_C$, $\boldsymbol{\delta}_{B_1}$ and $\boldsymbol{\delta}_{C_1}$ are perturbations of appropriate sizes that represent the learning errors.

Lemma 1. *Consider a zero order controller $\widetilde{K}$ (2) derived from the learned system $\widetilde{\Sigma}$. Set*

$$\boldsymbol{\xi} = \boldsymbol{\delta}_A + \widetilde{\boldsymbol{B}}\boldsymbol{K}_D\widetilde{\boldsymbol{C}} - \boldsymbol{B}\boldsymbol{K}_D\boldsymbol{C} \in \mathbb{R}^{n_x \times n_x}$$

and assume the following two conditions hold:

(a) The vector $\operatorname{vec}(\boldsymbol{\xi})$ *is in the column span of* $\widetilde{\boldsymbol{C}}_1^T \otimes \widetilde{\boldsymbol{B}}_1$;

(b) $\operatorname{span}(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1) \subseteq \operatorname{span}(\widetilde{\boldsymbol{C}}_1^T \otimes \widetilde{\boldsymbol{B}}_1)$.

Then, for any feedback matrix $\boldsymbol{\Delta} \in \mathbb{R}^{n_w \times n_z}$, there exists a matrix $\widetilde{\boldsymbol{\Delta}} \in \mathbb{R}^{n_w \times n_z}$ such that

$$\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) = \boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta}) , \tag{8}$$

where $\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}})$ is the system matrix of the closed-loop learned system (6) with feedback $\widetilde{\boldsymbol{\Delta}}$ and $\boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta})$ is the system matrix of the closed-loop true system with learned controller $\widetilde{K}$ and feedback $\boldsymbol{\Delta}$. Furthermore, the feedback matrix $\widetilde{\boldsymbol{\Delta}}$ can be decomposed as $\widetilde{\boldsymbol{\Delta}} = \widetilde{\boldsymbol{\Delta}}_0 + \widetilde{\boldsymbol{\Delta}}_1$ with

$$\widetilde{\boldsymbol{\Delta}}_0 = -\widetilde{\boldsymbol{B}}_1^+ \boldsymbol{\xi} \widetilde{\boldsymbol{C}}_1^+ \quad \text{and} \quad \widetilde{\boldsymbol{\Delta}}_1 = \widetilde{\boldsymbol{B}}_1^+ \boldsymbol{B}_1 \boldsymbol{\Delta} \boldsymbol{C}_1 \widetilde{\boldsymbol{C}}_1^+ , \tag{9}$$

where the superscript $^+$ denotes the Moore-Penrose inverse of the corresponding matrix.

Proof. Consider the least-squares problem

$$\underset{\operatorname{vec}(\widetilde{\boldsymbol{\Delta}})}{\arg\min} \|\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) - \boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta})\|_F^2$$

with the loss matrix

$$\boldsymbol{\mathcal{L}} = \widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) - \boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta}) = \begin{bmatrix} \boldsymbol{\mathcal{L}}_{11}(\widetilde{\boldsymbol{\Delta}}) & \boldsymbol{\mathcal{L}}_{12} \\ \boldsymbol{\mathcal{L}}_{21} & \boldsymbol{\mathcal{L}}_{22} \end{bmatrix} \tag{10}$$

$$= \begin{bmatrix} \boldsymbol{\xi} + \widetilde{\boldsymbol{B}}_1 \widetilde{\boldsymbol{\Delta}} \widetilde{\boldsymbol{C}}_1 - \boldsymbol{B}_1 \boldsymbol{\Delta} \boldsymbol{C}_1 & \boldsymbol{\delta}_B \widetilde{\boldsymbol{K}}_C \\ \widetilde{\boldsymbol{K}}_B \boldsymbol{\delta}_C & \boldsymbol{0} \end{bmatrix} . \tag{11}$$

In the loss matrix $\boldsymbol{\mathcal{L}}$ only the block $\boldsymbol{\mathcal{L}}_{11}$ depends on $\widetilde{\boldsymbol{\Delta}}$ and thus to find $\widetilde{\boldsymbol{\Delta}}$ that minimizes $\|\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) - \boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta})\|_F^2$, it is sufficient to consider

$$\underset{\operatorname{vec}(\widetilde{\boldsymbol{\Delta}})}{\arg\min} \|\boldsymbol{\mathcal{L}}_{11}(\widetilde{\boldsymbol{\Delta}})\|_F^2 . \tag{12}$$

Rewriting $\boldsymbol{\mathcal{L}}_{11}(\widetilde{\boldsymbol{\Delta}})$ as a vector gives

$$\operatorname{vec}(\boldsymbol{\mathcal{L}}_{11}(\widetilde{\boldsymbol{\Delta}})) = \boldsymbol{P}\operatorname{vec}\left(\widetilde{\boldsymbol{\Delta}}\right) - \left(\left(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1\right)\operatorname{vec}(\boldsymbol{\Delta}) - \operatorname{vec}(\boldsymbol{\xi})\right) , \tag{13}$$

with $\boldsymbol{P} = \widetilde{\boldsymbol{C}}_1^T \otimes \widetilde{\boldsymbol{B}}_1$. Then, the linear least-squares problem (12) has solution

$$\operatorname{vec}\left(\widetilde{\boldsymbol{\Delta}}^*\right) = \boldsymbol{P}^+ \left(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1\right)\operatorname{vec}(\boldsymbol{\Delta}) - \boldsymbol{P}^+ \operatorname{vec}(\boldsymbol{\xi}) , \tag{14}$$

where $\boldsymbol{P}^+$ is the Moore-Penrose inverse of $\boldsymbol{P}$. Because of assumption (a), $\operatorname{vec}(\boldsymbol{\xi})$ is in the column span of $\boldsymbol{P}$ and thus there exists $\boldsymbol{z}_1$ such that $\operatorname{vec}(\boldsymbol{\xi}) = \boldsymbol{P}\boldsymbol{z}_1$; similarly, because of assumption (b) there exists a $\boldsymbol{z}_2$ such that $(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1)\operatorname{vec}(\boldsymbol{\Delta}) = \boldsymbol{P}\boldsymbol{z}_2$. Thus, conditions (a) and (b) imply the existence of a vector $\boldsymbol{z}$ such that $(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1)\operatorname{vec}(\boldsymbol{\Delta}) - \operatorname{vec}(\boldsymbol{\xi}) = \boldsymbol{P}\boldsymbol{z}$ (where $\boldsymbol{z} = \boldsymbol{z}_2 - \boldsymbol{z}_1$). Therefore, the solution $\widetilde{\boldsymbol{\Delta}}^*$ achieves $\|\boldsymbol{\mathcal{L}}_{11}(\widetilde{\boldsymbol{\Delta}}^*)\|_F^2 = 0$. Notice that $\widetilde{\boldsymbol{\Delta}}^*$ is the minimal norm solution if the least-squares problem is underdetermined.

The off-diagonal block matrices $\boldsymbol{\mathcal{L}}_{12}$ and $\boldsymbol{\mathcal{L}}_{21}$ are zero because we assumed that $\widetilde{K}$ has order zero, and finally, the decomposition (9) of $\widetilde{\boldsymbol{\Delta}}^*$ follows from (14). □

Equation (8) shows that under the assumptions of Lemma 1, there exists a feedback matrix $\widetilde{\boldsymbol{\Delta}}$ such that the closed-loop learned system behaves as the closed-loop true system, which will allow us to draw conclusions about the true system based on the learned system; see Section 3.2. The decomposition (9) shows that the feedback matrix $\widetilde{\boldsymbol{\Delta}}$ is a sum of a matrix that depends on the learning error $\boldsymbol{\xi}$ and another matrix that depends on the feedback $\boldsymbol{\Delta}$ applied to the true system.

Corollary 1. *Consider a controller $\widetilde{K}$ as in* (2) *(not necessarily a zero order controller) derived from the learned system $\widetilde{\Sigma}$. Assume that $\boldsymbol{\delta}_B$ and $\boldsymbol{\delta}_C$ are both* $\mathbf{0}$*; and that conditions (a) and (b) in Lemma* 1 *hold. Then, for any feedback matrix $\boldsymbol{\Delta} \in \mathbb{R}^{n_w \times n_z}$, there exists a matrix $\widetilde{\boldsymbol{\Delta}} \in \mathbb{R}^{n_w \times n_z}$ such that* (8) *holds with the decomposition* (9)*.*

Proof. The proof of Corollary 1 follows the same steps in the proof of Lemma 1, except for the final step involving the off diagonal block matrices $\boldsymbol{\mathcal{L}}_{12}$ and $\boldsymbol{\mathcal{L}}_{21}$ from (10). However, since $\boldsymbol{\delta}_B$ and $\boldsymbol{\delta}_C$ are $\mathbf{0}$, the off diagonal terms $\boldsymbol{\mathcal{L}}_{12}$ and $\boldsymbol{\mathcal{L}}_{21}$ are also $\mathbf{0}$. □

Remark 1. *Conditions (a) and (b) of Lemma* 1 *hold if $\widetilde{\boldsymbol{B}}_1$ and $\widetilde{\boldsymbol{C}}_1$ of the learned system $\widetilde{\Sigma}$ are square full-rank matrices.*

3.2 Bounding the stability radius

Theorem 1. *Assume that either $\widetilde{K}$ is a zero order controller and the conditions of Lemma* 1 *hold or that $\widetilde{K}$ has a higher order and Corollary* 1 *applies. Then, the stability radius γ of the controlled true system $\Sigma_{\widetilde{K}}$ is lower bounded as $\gamma \geq \underline{\gamma}$ with*

$$\underline{\gamma} = \frac{1}{\|\boldsymbol{B}_1\|_2 \|\boldsymbol{C}_1\|_2} \left(\widetilde{\gamma} \cdot \frac{1}{\left\|\widetilde{\boldsymbol{B}}_1^+\right\|_2 \left\|\widetilde{\boldsymbol{C}}_1^+\right\|_2} - \|\boldsymbol{\xi}\|_2 \right) , \tag{15}$$

where $\widetilde{\gamma}$ is the stability radius of the controlled learned systems $\widetilde{\Sigma}_{\widetilde{K}}$ as defined in (7)*.*

Proof. Since either the conditions of Lemma 1 or Corollary 1 hold, the closed-loop true system and closed-loop learned system are equivalent in the sense that $\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}}) = \boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta})$ with $\widetilde{\boldsymbol{\Delta}} = \widetilde{\boldsymbol{\Delta}}_0 + \widetilde{\boldsymbol{\Delta}}_1 = -\widetilde{\boldsymbol{B}}_1^+ \boldsymbol{\xi} \widetilde{\boldsymbol{C}}_1^+ + \widetilde{\boldsymbol{B}}_1^+ \boldsymbol{B}_1 \boldsymbol{\Delta} \boldsymbol{C}_1 \widetilde{\boldsymbol{C}}_1^+$. The quantities $\widetilde{\boldsymbol{\Delta}}_0$ and $\widetilde{\boldsymbol{\Delta}}_1$ are defined in (9). Thus, the closed-loop learned system matrix $\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}})$ is stable if and only if the closed-loop true system matrix $\boldsymbol{A}_{\widetilde{K}}(\boldsymbol{\Delta})$ is also stable.

The stability radius of the controlled learned system $\widetilde{\Sigma}_{\widetilde{K}}$ is $\widetilde{\gamma}$, so we require $\|\widetilde{\boldsymbol{\Delta}}\|_2 \leq \widetilde{\gamma}$ for $\widetilde{\boldsymbol{A}}_{\widetilde{K}}(\widetilde{\boldsymbol{\Delta}})$ to be stable. By the equivalency of the closed-loop true system and closed-loop learned system, additionally substituting $\widetilde{\boldsymbol{\Delta}}_0$ and $\widetilde{\boldsymbol{\Delta}}_1$ as defined in (9), we obtain

$$\|\widetilde{\boldsymbol{\Delta}}\|_2 \leq \|\widetilde{\boldsymbol{B}}_1^+ \boldsymbol{\xi} \widetilde{\boldsymbol{C}}_1^+\|_2 + \|\widetilde{\boldsymbol{B}}_1^+ \boldsymbol{B}_1 \boldsymbol{\Delta} \boldsymbol{C}_1 \widetilde{\boldsymbol{C}}_1^+\|_2 .$$

Thus, to have $\|\widetilde{\boldsymbol{\Delta}}\|_2 \leq \widetilde{\gamma}$, we want to derive $\boldsymbol{\Delta}$ such that

$$\|\widetilde{\boldsymbol{B}}_1^+\|_2 \|\boldsymbol{\xi}\|_2 \|\widetilde{\boldsymbol{C}}_1^+\|_2 + \|\widetilde{\boldsymbol{B}}_1^+\|_2 \|\boldsymbol{B}_1\|_2 \|\boldsymbol{\Delta}\|_2 \|\boldsymbol{C}_1\|_2 \|\widetilde{\boldsymbol{C}}_1^+\|_2 \leq \widetilde{\gamma} ,$$

which is satisfied if $\|\boldsymbol{\Delta}\|_2 \leq \underline{\gamma}$ with $\underline{\gamma}$ defined in (15). This means, if $\|\boldsymbol{\Delta}\|_2 \leq \underline{\gamma}$ holds, then the closed-loop true system is stable because the closed-loop learned system with feedback $\widetilde{\boldsymbol{\Delta}}$ is stable and it is equivalent to the closed-loop true systems. □

If $\boldsymbol{B}_1$ has more rows than columns and $\boldsymbol{C}_1$ has more columns than rows, i.e., the dimension of of inputs and outputs is smaller than the state dimension, then a left and right, respectively, pseudo inverse exist. If further $\boldsymbol{\delta}_B, \boldsymbol{\delta}_C, \boldsymbol{\delta}_{B_1}, \boldsymbol{\delta}_{C_1}$ are zero matrices, then $\widetilde{\boldsymbol{\Delta}}$ with the decomposition (9) simplifies to $\widetilde{\boldsymbol{\Delta}} = \widetilde{\boldsymbol{\Delta}}_0 + \boldsymbol{\Delta}$ and avoids the matrix-matrix multiplication $\widetilde{\boldsymbol{B}}_1^+ \boldsymbol{B}_1$ and $\boldsymbol{C}_1 \widetilde{\boldsymbol{C}}_1^+$.

4 Sample complexity of learning a controller from a system identified with least-squares regression

We now study a least-squares regression procedure for learning a model of the state transition dynamics. We assume knowledge of the relationship between states and inputs/outputs: we have the matrices $\boldsymbol{B}, \boldsymbol{B}_1, \boldsymbol{C}, \boldsymbol{C}_1$ of a realization of the true system and so obtain a state-space model in the form (1) of the learned system $\widetilde{\Sigma}$ with $\boldsymbol{\delta}_B = \mathbf{0}, \boldsymbol{\delta}_{B_1} = \mathbf{0}, \boldsymbol{\delta}_C = \mathbf{0}, \boldsymbol{\delta}_{C_1} = \mathbf{0}$.

First, excite the true system Σ at normally distributed disturbances $\boldsymbol{w}^{(1)}, \dots, \boldsymbol{w}^{(N)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{S}_w)$, inputs $\boldsymbol{u}^{(1)}, \dots, \boldsymbol{u}^{(N)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{S}_u)$, and initial conditions $\boldsymbol{x}^{(1)}(0), \dots, \boldsymbol{x}^{(N)}(0) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{S}_x)$ with covariance matrices $\boldsymbol{S}_w \in \mathbb{R}^{n_w \times n_w}, \boldsymbol{S}_u \in \mathbb{R}^{n_u \times n_u}$, and $\boldsymbol{S}_x \in \mathbb{R}^{n_x \times n_x}$, respectively. The corresponding velocities are $\dot{\boldsymbol{x}}^{(1)}(0), \dots, \dot{\boldsymbol{x}}^{(N)}(0)$, which we measure to obtain the observed velocities $\dot{\boldsymbol{x}}_\epsilon^{(1)}, \dots, \dot{\boldsymbol{x}}_\epsilon^{(N)}$ with measurement error

$$\dot{\boldsymbol{x}}_\epsilon^{(i)} = \dot{\boldsymbol{x}}^{(i)}(0) + \epsilon^{(i)} ,$$

where $\epsilon^{(1)}, \ldots, \epsilon^{(N)}$ are independent zero mean Gaussian random vectors with covariance matrix $\boldsymbol{S}_\epsilon \in \mathbb{R}^{n_x \times n_x}$. If the velocities cannot be observed directly, then they can be approximated by simulating the true system, recording states, and approximating the time derivatives. Note that rather than simulating a single trajectory, we collect information from the initial step of N independent trajectories.

We obtain matrices $\boldsymbol{X}$, $\dot{\boldsymbol{X}}_\epsilon$, $\boldsymbol{U}$, $\boldsymbol{W}$ by concatenating the N samples as $\dot{\boldsymbol{X}}_\epsilon = [\dot{\boldsymbol{x}}_\epsilon^{(1)}, \ldots, \dot{\boldsymbol{x}}_\epsilon^{(N)}]$, $\boldsymbol{X} = [\boldsymbol{x}^{(1)}(0), \ldots, \boldsymbol{x}^{(N)}(0)]$, $\boldsymbol{W} = [\boldsymbol{w}^{(1)}, \ldots, \boldsymbol{w}^{(N)}]$, $\boldsymbol{U} = [\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(N)}]$, and $\boldsymbol{\varepsilon} = [\epsilon^{(1)}, \ldots, \epsilon^{(N)}]$. The system matrix $\widetilde{\boldsymbol{A}}$ of the learned system $\widetilde{\Sigma}$ is then the least-squares estimator

$$\widetilde{\boldsymbol{A}} = \left(\dot{\boldsymbol{X}}_\epsilon - \boldsymbol{B}\boldsymbol{U} - \boldsymbol{B}_1\boldsymbol{W}\right)\boldsymbol{X}^T\left(\boldsymbol{X}\boldsymbol{X}^T\right)^{-1}, \quad (16)$$

which is a random matrix. Consequently, the learning error $\boldsymbol{\delta}_A$ is a random matrix too.

Theorem 2. *Suppose every realization of the learning error $\boldsymbol{\delta}_A$ satisfies $\operatorname{vec}(\boldsymbol{\delta}_A) \in \operatorname{span}(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1)$. Let the number of samples N satisfy $N \geq 8n_x + 16\log(2/\tau)$, where $\tau \in (0,1)$ is a failure probability. Then, the stability radius γ of $\Sigma_{\widetilde{K}}$ (the controlled true system with the learned controller) is lower bounded as $\gamma \geq \underline{\gamma}$ with probability at least $1-\tau$, where*

$$\underline{\gamma} = \frac{\widetilde{\gamma}}{\kappa(\boldsymbol{B}_1)\kappa(\boldsymbol{C}_1)} - \frac{\beta}{\|\boldsymbol{B}_1\|_2\|\boldsymbol{C}_1\|_2}\sqrt{\frac{\log(18/\tau)}{N}}, \quad (17)$$

with $\beta = 16\sqrt{2}\|\boldsymbol{S}_\varepsilon\|_2^{1/2}\|\boldsymbol{S}_x^{-1/2}\|_2\sqrt{n_x}$, $\widetilde{\gamma}$ the stability radius of the controlled learned system $\widetilde{\Sigma}_{\widetilde{K}}$, and $\kappa(\cdot)$ denoting the condition number in the $\|\cdot\|_2$ norm of the matrix argument.

Proof. We derive the stated bound for $\underline{\gamma}$ by bounding $\|\boldsymbol{\delta}_A\|_2$, then substituting the bound directly into the result of Theorem 1. Consider the least squares estimator $\widetilde{\boldsymbol{A}}$ defined in (16). By the definition of the model,

$$\dot{\boldsymbol{X}}_\epsilon - \boldsymbol{B}\boldsymbol{U} - \boldsymbol{B}_1\boldsymbol{W} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{\varepsilon},$$

so the learning error $\boldsymbol{\delta}_A$ is given by $\boldsymbol{\delta}_A = \widetilde{\boldsymbol{A}} - \boldsymbol{A} = \boldsymbol{\varepsilon}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{X}^T)^{-1}$. Note that since $\boldsymbol{X}$ is a random matrix of column vectors from $\mathcal{N}(0, \boldsymbol{S}_x)$ and because $N > n_x$, the matrix $\boldsymbol{X}\boldsymbol{X}^T$ is invertible with probability 1. Lemma 3 in [9] states the following: If $\boldsymbol{x}^{(i)}(0) \sim \mathcal{N}(0, \boldsymbol{S}_x)$ (and $\boldsymbol{S}_x$ invertible), $\boldsymbol{\varepsilon} \in \mathbb{R}^{n_x \times N}$ with $\varepsilon^{(i)} \in \mathcal{N}(0, \boldsymbol{S}_\varepsilon)$ for $i = 1, \ldots, N$ and if $N \geq 8n_x + 16\log(2/\tau)$ then

$$\|\boldsymbol{Y}\|_2 \leq 16\|\boldsymbol{S}_\varepsilon\|_2^{1/2}\|\boldsymbol{S}_x^{-1/2}\|_2\sqrt{\frac{2n_x\log(18/\tau)}{N}}$$

with probability at least $1-\delta$, where $\boldsymbol{Y} = (\boldsymbol{X}\boldsymbol{X}^T)^{-1}\boldsymbol{X}\boldsymbol{\varepsilon}^T$. Thus, the lemma applies for $\boldsymbol{Y} = \boldsymbol{\delta}_A^T$.

It was assumed that $\operatorname{vec}(\boldsymbol{\delta}_A) \in \operatorname{span}(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1)$, which implies condition (a) of Lemma 1. Condition (b) of Lemma 1 holds since $\widetilde{\boldsymbol{B}}_1 = \boldsymbol{B}_1$ and $\widetilde{\boldsymbol{C}}_1 = \boldsymbol{C}_1$. Furthermore $\boldsymbol{\delta}_B = \boldsymbol{0}$ and $\boldsymbol{\delta}_C = \boldsymbol{0}$ and thus Corollary 1 applies too, meaning that Theorem 1 applies independent of whether we have a zero or higher-order controller. Now let $\beta = 16\|\boldsymbol{S}_\varepsilon\|_2^{1/2}\|\boldsymbol{S}_x^{-1/2}\|_2\sqrt{2n_x}$ and substitute the bound for $\|\boldsymbol{\delta}_A\|_2$ into the result of Theorem 1, additionally noting that $\widetilde{\boldsymbol{B}}_1 = \boldsymbol{B}_1$ and $\widetilde{\boldsymbol{C}}_1 = \boldsymbol{C}_1$ to obtain the stated bound $\underline{\gamma}$ in (17). □

Remark 2. *In case of a single-input-single-output system, the condition numbers $\kappa(\boldsymbol{B}_1)$ and $\kappa(\boldsymbol{C}_1)$ in (17) are 1.*

5 Numerical Results

We consider data generated from four dynamical systems, which are listed in Table 1. All these systems are motivated by applications in aerospace engineering and can be obtained from the cited reference in Table 1.

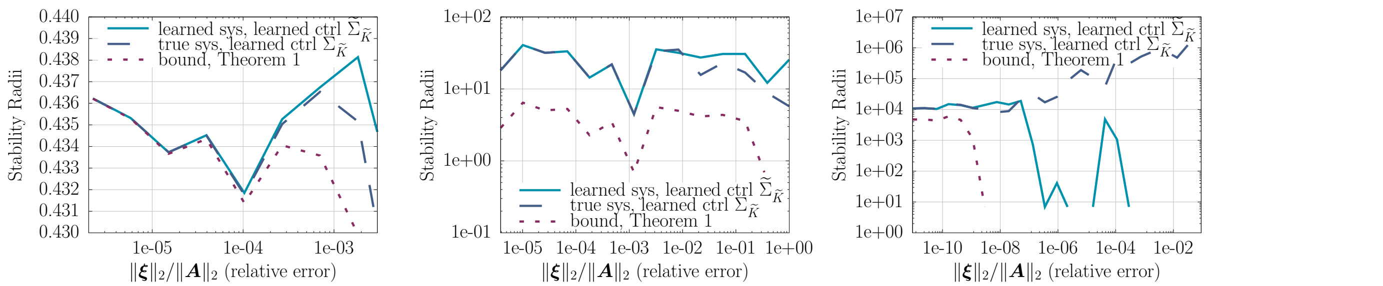
5.1 Efficiency of the proposed stability bound

Let Σ be either the HE2, JE2, or BE747 system; cf. Table 1. We consider learned systems $\widetilde{\Sigma}$ that are obtained by perturbing the system matrices of the true system Σ. We select a random matrix $\boldsymbol{\delta}_A$ by constructing an orthonormal basis for $\operatorname{span}(\boldsymbol{C}_1^T \otimes \boldsymbol{B}_1)$ and selecting independent standard normal coefficients for forming a linear combination with the bases vectors. The corresponding random matrix $\boldsymbol{\delta}_A$ is then scaled to have magnitude in the range of 10^{-5} to 2.6102, 10^{-5} to 1, and 10^{-5} to 10^5 for systems HE2, JE2, BE747, respectively. All other errors $\boldsymbol{\delta}_B, \boldsymbol{\delta}_C, \boldsymbol{\delta}_{B_1}, \boldsymbol{\delta}_{C_1}$ are $\boldsymbol{0}$. We then apply HIFOO [14, 3] with the $\mathcal{H}_\infty$ routine to compute a controller $\widetilde{K}$ of order $\max\{n_u, n_y\}$. We compute the stability radius $\widetilde{\gamma}$ of the controlled learned system $\widetilde{\Sigma}_{\widetilde{K}}$ and the bound $\underline{\gamma}$. We also compute the stability radius γ of the controlled true system $\Sigma_{\widetilde{K}}$ as a benchmark.

Figure 1a shows the stability radii γ and $\widetilde{\gamma}$ as well as the bound $\underline{\gamma}$ versus the relative learning error $\|\boldsymbol{\xi}\|_2/\|\boldsymbol{A}\|_2$ for the HE2 system. The learned HE2 system with a higher order controller satisfies the conditions of Corollary 1. Hence, the results demonstrate

Name	Description	n_x	n_u	n_y	n_w	n_z	$\|B_1^+\|_2$	$\|C_1^+\|_2$	$\|C_1B_1\|_2$
HE2	Apache AH-64 helicopter [16]	4	2	2	4	4	1	1	1
JE2	Rolls Royce Spey jet engine [16]	21	3	3	3	3	2.5149×10^3	2.9347×10^3	1.5737×10^{-7}
BE747	Damaged Boeing 747 aircraft [18]	5	5	5	2	2	10.5504	2.000	0
UAV	Raptor 90 RC helicopter [10]	10	4	5	2	5	2.7318	1	0.6529

Table 1: Systems used in the numerical experiments.



(a) HE2 (b) BE747 (c) JE2

Figure 1: Plot (a) shows for the HE2 system that the bound of Theorem 1 closely lower bounds the stability radius corresponding to the learned controller: a relative error of 0.2% leads to an underestimate in the stability radius of 1%. The efficiency of the lower bound (15) varies depending on system properties. For the BE747 system, plot (b) shows that the bound is informative up to a relative error of 30%. In contrast, for JE2, the bound is no longer informative after a relative error of just 10^{-9} as shown in plot (c).

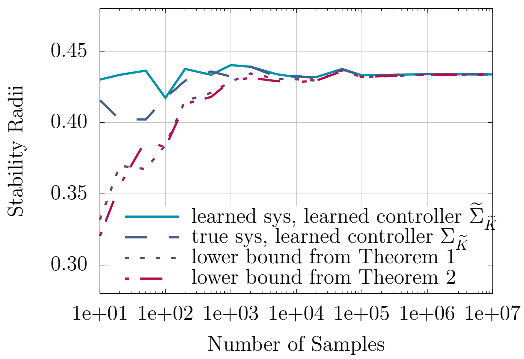
Theorem 1 and that the bound $\underline{\gamma}$ is efficient for a large range of learning errors $\|\boldsymbol{\xi}\|_2/\|\boldsymbol{A}\|_2$.

The stability radii and bounds corresponding to systems BE747 and JE2 are shown in Figure 1b-c. Both systems satisfy the assumptions of Corollary 1. The proposed bound $\underline{\gamma}$ is efficient for the BE747 system for large errors of up to 1% and is informative, i.e., positive, for errors up to 30%. In contrast, the bound for the JE2 system does not provide an informative stability radius beyond a relative error of 10^{-9}. The proposed bound's quality depends on properties of the system and its realization. For example, we use a realization of the JE2 system that has large norms $\|\boldsymbol{B}_1^+\|_2$ and $\|\boldsymbol{C}_1^+\|_2$, which means that in the proposed bound (15), the first term $\widetilde{\gamma}/(\|\boldsymbol{B}_1^+\|_2\|\boldsymbol{C}_1^+\|_2)$ is small so even a small error $\|\boldsymbol{\xi}\|_2$ can make the lower bound negative and thus uninformative. This is in contrast to the BE747 and HE2 systems that have norms $\|\boldsymbol{B}_1^+\|$ and $\|\boldsymbol{C}_1^+\|$ orders of magnitude smaller than the JE2 system.

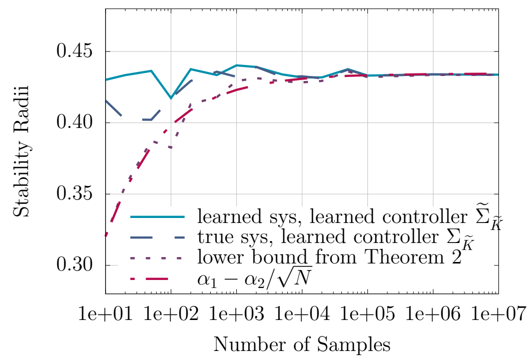
Note that the realization independent Markov parameter $\|\boldsymbol{C}_1\boldsymbol{B}_1\|_2$ appears to coincide with the quality of bound. For the HE2 system, the Markov parameter is 1 and the bound $\underline{\gamma}$ closely bounds the true stability radius. On the other hand, when the Markov parameter is close to 0 for both the BE747 and JE2 systems, the bound is of lower quality, as it underestimates by an order of magnitude for the BE747 system while it is only informative for the JE2 system up to a relative error of 10^{-9}. It remains future work to further analyze the dependence of the bound on the specific realization (such as the norms $\|\boldsymbol{B}_1^+\|$ and $\|\boldsymbol{C}_1^+\|$) and derive the bound fully in terms of system properties instead (such as the first Markov parameter $\|\boldsymbol{C}_1\boldsymbol{B}_1\|_2$ as motivated by the numerical results).

5.2 Learning model via least-squares regression

We now apply the least-squares approach discussed in Section 4 to the HE2 system to obtain the learned system $\widetilde{\Sigma}$. The covariance matrices for sampling and noise are set to be diagonal matrices with all diagonal entries a constant σ, denoted $\boldsymbol{S}(\sigma)$. The covariance matrices are then $\boldsymbol{S}_x = \boldsymbol{S}(2)$, $\boldsymbol{S}_u = \boldsymbol{S}(1)$, $\boldsymbol{S}_w = \boldsymbol{S}(1/2)$, $\boldsymbol{S}_\varepsilon = \boldsymbol{S}(1/5)$. Figure 2 shows a trade off between number of samples N (learning costs) and stability radius; one order of magnitude reduction in the number of samples results in less than 5% reduction in stability radius in this example. In other words, with a context-aware goal in mind, the learn-



(a) lower bounds from Theorems 1 and 2



(b) lower bound w.r.t. $\alpha_1 - \alpha_2/\sqrt{N}$

Figure 2: HE2: Plot (a) shows that the lower bound from Theorem 2 is close to the lower bound from Theorem 1, which indicates that the bound of Theorem 2 based on the number of samples does not add additional conservatism in this example. Plot (b) demonstrates that the bound of Theorem 2 behaves as expected as $\alpha_1 - \alpha_2/\sqrt{N}$ with constants $\alpha_1, \alpha_2 > 0$.

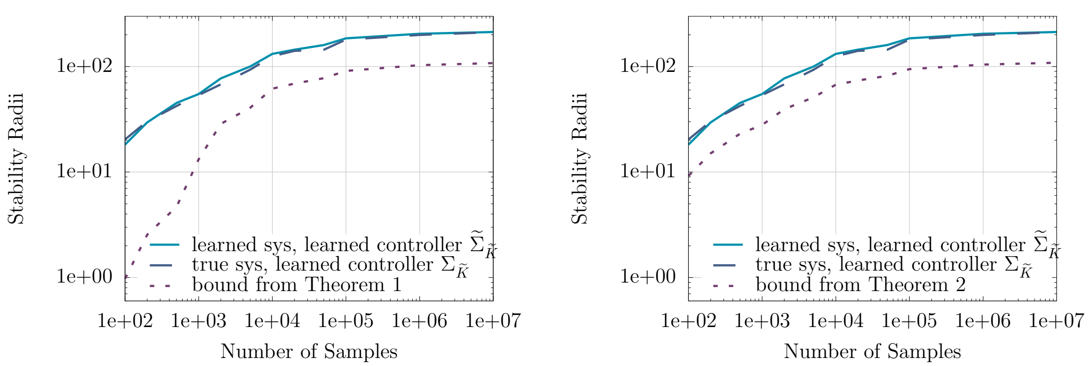


Figure 3: UAV: The results show that the bounds derived in Theorem 1 (left) and Theorem 2 (right) become efficient indicators of the stability radius in this example when the assumptions of the theorems are not met. The indicators underestimate the true stability radius by about 50% but are within the same order of magnitude.

ing costs could be reduced significantly without losing substantial accuracy in the eventual control goal.

We now apply the same least-squares regression procedure to the UAV system and show the stability radius versus increasing samples. Unlike the systems HE2, BE747, JE2, the UAV system violates condition (b) in Lemma 1. However, Figure 3 suggests that the bounds derived in Theorem 1 and Theorem 2 are still efficient indicators in practice.

Acknowledgements

The first and third author acknowledge partial support of the NSF under Grant No. 2012250. The third author acknowledges support from the AFOSR award FA9550-21-1-0222 (Dr. Fariba Fahroo). The second author acknowledge the partial support of of the NSF under Grant No. 1819110.

References

[1] T. Alsup and B. Peherstorfer. Context-aware surrogate modeling for balancing approximation and sampling costs in multi-fidelity importance sampling and Bayesian inverse problems. *arXiv:2010.11708*, 2020.

[2] A. C. Antoulas and B. D. O. Anderson. On the scalar rational interpolation problem. *IMA Journal of Mathematical Control & Information*, 3(2-3):61–88, 1986.

[3] D. Arzelier, D. Georgia, S. Gumussoy, and D. Henrion. H2 for HIFOO. *arXiv preprint arXiv:1010.1442*, 2010.

[4] C. Audouze, F. De Vuyst, and P. B. Nair. Reduced-order modeling of parameterized PDEs using time–space-parameter principal component analysis. *Int J Numer Methods Eng*, 80(8):1025–1057, 2009.

[5] C. Beattie and S. Gugercin. Realization-independent $\mathcal{H}_2$-approximation. In *Proc. IEEE Conf. Decis. Control*, pages 4953–4958, Maui, HI, USA, 2012.

[6] P. Benner. Solving large-scale control problems. *IEEE Control Systems Magazine*, 24(1):44–59, 2004.

[7] P. Benner, J.-R. Li, and T. Penzl. Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems. *Numerical Linear Algebra with Applications*, 15(9):755–777, 2008.

[8] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc.Natl.Acad.Sci.*, 113(15):3932–3937, 2016.

[9] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, pages 1–47, 2019.

[10] J. Gadewadikar, F. Lewis, K. Subbarao, and B. M. Chen. Structured H-Infinity command and control-loop design for unmanned helicopters. *J Guid Control Dyn*, 31(4):1093–1102, 2008.

[11] I. V. Gosea and A. C. Antoulas. Data-driven model order reduction of quadratic-bilinear systems. *Numerical Linear Algebra with Applications*, 25(6):e2200, 2018.

[12] I. V. Gosea, S. Gugercin, and C. Beattie. Data-driven balancing of linear dynamical systems. *arXiv preprint arXiv:1010.1442*, 2021.

[13] S. Gugercin and A. C. Antoulas. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748–766, 2004.

[14] S. Gumussoy, D. Henrion, M. Millstone, and M. L. Overton. Multiobjective robust control with HIFOO 2.0. *IFAC Proceedings Volumes*, 42(6):144–149, 2009.

[15] J. S. Hesthaven and S. Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018.

[16] F. Leibfritz. COMPLeIB, COnstraint Matrix-optimization Problem LIbrary-a collection of test examples for nonlinear semidefinite programs, control system design and related problems. *Dept. Math., Univ. Trier, Trier, Germany, Tech. Rep*, 2004, 2004.

[17] A. J. Mayo and A. C. Antoulas. A framework for the solution of the generalized realization

problem. *Linear Algebra and its Applications*, 425(2–3):634–662, 2007.

[18] M. Ornik, A. Israel, and U. Topcu. Control-oriented learning on the fly. *arXiv:1709.04889*, 2017.

[19] B. Peherstorfer. Multifidelity Monte Carlo estimation with adaptive low-fidelity models. *SIAM/ASA Journal on Uncertainty Quantification*, 7:579–603, 2019.

[20] B. Peherstorfer and K. Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196–215, 2016.

[21] E. Qian, B. Kramer, B. Peherstorfer, and K. Willcox. Lift & learn: Physics-informed machine learning for large-scale nonlinear dynamical systems. *Physica D: Nonlinear Phenomena*, 406, 2020.

[22] T. Reis and T. Stykel. Stability analysis and model order reduction of coupled systems. *Mathematical and Computer Modelling of Dynamical Systems*, 13(5):413–436, 2007.

[23] T. Reis and T. Stykel. A survey on model reduction of coupled systems. In *Model order reduction: theory, research aspects and applications*, pages 133–155. Springer, 2008.

[24] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.

[25] R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox. Projection-based model reduction: Formulations for physics-based machine learning. *Computers & Fluids*, 179:704–717, 2019.

[26] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz. On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics*, 1(2):391–421, 2014.

[27] S. Tu, R. Boczar, A. Packard, and B. Recht. Non-asymptotic analysis of robust control from coarse-grained identification. *arXiv:1707.04791*, 2017.

[28] A. Varga and B. D. Anderson. Accuracy-enhancing methods for balancing-related frequency-weighted model and controller reduction. *Automatica*, 39(5):919–927, 2003.