Robust Data-Driven Control with Noisy Data

Chin-Yao Chang and Andrey Bernstein

Abstract—This paper presents a robust data-driven controller design based on the noisy input-output data without assumptions on the statistical properties of the noises. We start with the direct data-representation of system models that take elements from behavioral system theory, followed by analyses of the upper bound of the "modeling" error with the data representation with presence of noises. Some pre-conditioning methods are put into the context based on how the derived bound is structured. We lastly leverage the upper bound to develop robust controllers that ride through the data noises.

I. Introduction

In the last decade, there has been a major surge of interest in machine learning (ML) because it can find reasonable solutions to challenging and practical optimization problems which were otherwise unsolvable with standard optimization approaches. One can view ML as black-box identification of system models or optimal policies by using massive amounts of input-output data. Due to the lack of prior knowledge of targeted systems, artificial neural network (ANN) model structure has been widely adapted and received great success. However, there remain many applications (e.g. power systems) where partial system information is known in advance and the main challenges are instead on the robustness and resilience against all sorts of disturbances such as noisy data. We therefore put efforts on adding insights on robust controls with noisy data in this work.

The measured input-output data are usually used for system identification. One branch of system identification is the direct data representation of dynamical system models which were explored in the 80's [2]–[5]. The popularity of these behavioral system approaches have been recently revived. [6] applies behavioral system theory to develop data-enabled predictive control (DeePC), which is a data-driven alternative to model predictive control (MPC). [7] draws connections between the behavioral system model and linear matrix inequalities (LMIs) stability analysis so that some classic controller designs can be applied to behavioral system models constructed from data. Series of follow-up works further expand these ideas to robust control [8], [9], switched linear systems with unknown

C.-Y, Chang, and A. Bernstein are with the National Renewable Energy Laboratory, Golden, CO 80401, USA (Emails: {chinyao.chang,andrey.bernstein}@nrel.gov).

This work was authored in part by NREL, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by DOE Office of Electricity, Advanced Grid Modeling Program, through agreement NO. 33652. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

switching [10], time-varying linear systems [11] and realtime applications [12], [13]. Fundamentally, behavioral system theory does not extract more information from data compared to classical system identification approaches. However, for controller design purposes, there are some subtle advantages of behavioral system theory-based approaches, detailed in [14].

Noisy data draws errors on the identified system and in turns the performance of the controllers built upon the model are compromised. Based on the disturbance types, various specialized methods are available, e.g., principal component analysis (PCA) [15], regularization [16], or methods for noisy labels [17]. In practice, it is very challenging to have prior knowledge on the noise or make proper assumptions on it. Both [8] and [1] are about robust data-driven predictive controls assuming no prior knowledge on the noise except an upper bound on the norm. [8] showed recursive feasibility and stability of the data-driven MPC formulated with the behavioral system model. [1] proposed a zonotopic data-driven predictive control scheme that guarantees robust constraint satisfaction. Our focus in this paper is not on the predictive controls, but on the characterization of the modeling errors propagated from the noises, followed by a robust closed-loop feedback gain design.

Contributions: Our first contribution is deriving an upper bound on data model representation errors that originate from disturbances of the collected data with no assumption on statistical properties of the noises. The upper bounds are conservative while we conjecture that there exists no much tighter bound from mathematical analysis because the condition number always shows up mechanically. Therefore, we next put some pre-conditioning methods in the context aiming to reduce the condition number. Although the main purpose of reducing the condition number is tightening the analytical upper bounds, numerical examples imply that minimizing the condition number of the data matrix could reduce actual modeling errors. We finally leverage the upper bounds for robust controller design for linear systems, with some additional comments on how those results can be extended to switched linear systems.

Notations: We denote by \mathbb{R} and \mathbb{R}_+ the sets of real and positive real numbers, respectively. For a matrix A, we write $\|A\|$ and $\|A\|_F$, respectively, as its 2-norm and Frobenius norm. The pseudo inverse of A is written as A^{\dagger} with the subscript of R or L to indicate the right or left inverse if necessary. A matrix is called standardized if every column of it has unit 2-norm.

II. DATA REPRESENTATION OF SYSTEM MODELING

In this section, we review the results in [7] that connect Willems *et al.*'s fundamental lemma for behavioral system theory and simple linear algebra, and then expand the discussion to switched linear systems.

A. Data representation of LTI systems

This subsection reviews some results in [7] that will be used for the remainder of the paper. Consider the following simplified LTI system:

$$x(k+1) = Ax(k) + Bu(k), \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, A and B are in proper dimensions. Here we assume that all the elements of x can be measured. Define the collection of the measured data for the time horizon T in the following:

$$U_0 = [u_d(0), \cdots, u_d(T-1)],$$
 (2a)

$$X_0 = [x_d(0), \cdots, x_d(T-1)],$$
 (2b)

$$X_1 = [x_d(1), \cdots, x_d(T)],$$
 (2c)

where $x_d(k)$ and $u_d(k)$ are, respectively, the state and control input data points at time k. If all the data points are noiseless, then by (1) and the definition of the data matrices in (2), we have

$$X_1 = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}. \tag{3}$$

Equation (3) indicates that as long as $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$ is right invertible, we can find the system matrices, A and B, straight from the data by

$$\begin{bmatrix} B & A \end{bmatrix} = X_1 \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^{\dagger}. \tag{4}$$

The full row rank (invertibility) of $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$ is formally stated in Assumption 1.

Assumption 1. (Full row rank of the data matrix).

$$rank\left(\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}\right) = m + n \tag{5}$$

As a side note, for the LTI system (1) expanded with y(k) = Cx(k) + Du(k) with only y being measured instead of x, a similar rank condition to (5) posed on a *Hankel matrix* constructed from u_d can lead to a similar pure data representation of the system behavior in the sense of input-output pairs (assuming the linear system is observable and controllable). The full rank condition and the data representation of the system behavior are respectively stated as persistently exciting condition and Willems $et\ al.$'s fundamental lemma. More details are in [6] and [7].

B. Switched linear systems with known modes

We consider the following switched linear system:

$$x(k+1) = A_{\sigma(k)}x(k) + B_{\sigma(k)}u(k),$$

$$\sigma(k) = f(x(k)),$$
(6)

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $\sigma \in \Gamma := [1, \cdots, \gamma]$ with some finite and known γ number of modes. If one follows standard steps in Section II-A, identifying the system matrices A_{σ} and B_{σ} requires the system to stay in the same mode long enough to construct data matrices as shown in (2). This restriction hinders the practicability of the data-driven control in the sense that the switching sequence might change frequently in practice. We show next that if the system mode at each time k is given, then even with arbitrary switching sequence, similar data matrices can still capture system matrices A_{σ} and B_{σ} for each individual mode.

Given the switch sequence $\sigma(0), \dots, \sigma(T)$, we enumerate the input and measured data by $(u_d(0), \dots, u_d(T-1))$ and $(x_d(0), \dots, x_d(T))$, and then construct the following data matrices for each mode i:

$$U_{i,0} = [u_{i,d}(0), \cdots, u_{i,d}(T-1)],$$

 $X_{i,0} = [x_{i,d}(0), \cdots, x_{i,d}(T-1)],$

where for every $k = 0, \dots, T$,

$$u_{i,d}(k) = \begin{cases} u_d(k) & \text{if } \sigma(k) = i \\ 0 & \text{otherwise} \end{cases},$$

$$x_{i,d}(k) = \begin{cases} x_d(k) & \text{if } \sigma(k) = i \\ 0 & \text{otherwise} \end{cases}.$$

The data matrices $U_{i,0}$ and $X_{i,0}$ are sparse by construction because column k has all zero elements if $\sigma(k) \neq i$. We next make the following mild assumption on $U_{i,0}$ and $X_{i,0}$:

Assumption 2. For each $i \in \Gamma$,

$$rank\left(\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}\right) = n + m. \tag{7}$$

Assumption 2 holds as long as T is long enough such that there are sufficient occurrences for every mode $i \in \Gamma$. The X_1 matrix is still defined in the same way as the LTI case: $X_1 = [x_d(1), \cdots, x_d(T)]$. The following theorem shows how the system matrices $[B_i \ A_i]$ for all $i \in \Gamma$ are identified.

Lemma II.1. If Assumption 2 holds, then for every $i \in \Gamma$, there exists $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^\dagger$ such that

$$[B_i \ A_i] = X_1 \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^{\dagger}, \quad \forall i \in \Gamma.$$
 (8)

Proof. There are infinite number of selections of the pseudo inverse of $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}$, while we choose the one with a structure such that (8) follows. Define $U_{i,0}^s$ and $X_{i,0}^s$ respectively as the sub-matrices of $U_{i,0}$ and $X_{i,0}$ that collect all the non-zero

columns of them. Given an arbitrary pseudo inverse of $\begin{bmatrix} U_{i,0}^s \\ X_{i,0}^s \end{bmatrix}$, we define $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^\dagger$ such that its rows associated with non-zero columns of $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}$ are comprised of $\begin{bmatrix} U_{i,0} \\ X_{i,0}^s \end{bmatrix}^\dagger$, and the remaining rows are all zero. This keeps the property that $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} \cdot \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^\dagger = I$, while more importantly,

$$\begin{bmatrix} U_{j,0} \\ X_{j,0} \end{bmatrix} \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^{\dagger} = 0, \quad \text{if } i \neq j.$$
 (9)

Equation (9) will be very useful in simplifying the following equality

$$X_1 = \sum_{j \in \Gamma} [B_j \ A_j] \begin{bmatrix} U_{j,0} \\ X_{j,0} \end{bmatrix}, \tag{10}$$

which is straight from (6). Equation (8) for each $i \in \Gamma$ follows by right multiplying (10) by $\begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix}^{\dagger}$.

Applying Lemma II.1, one can derive the modeling for each mode from data.

III. CONTROLLER DESIGN WITH NOISY DATA

In this section, we first derive a bound of the errors of data representation of system models. The bound is proportional to the condition number of the data matrix, so we extend some discussions on reducing the condition number. The improved condition number tightens the bound and increases the practicability of robust controller designs that leverage the bound.

A. Bounds for system identification errors

In this section, we consider LTI system (1) for simplicity. If Assumption 1 holds and the data are noisy, the linear model identified from (4) is not exactly the original $\begin{bmatrix} B & A \end{bmatrix}$. With presence of the inaccuracy of the data, (4) is rewritten with slight abuse of notations

$$[B^e \ A^e] = X_1 \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^{\dagger}, \quad [B \quad A] = X_1^{\star} \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix}_R^{\dagger}, \quad (11)$$

where $[B^e \ A^e]$ is the estimated system matrix. The actual system matrix, $[B \ A]$, can not be derived straight from the data because the accurate system values $(X_1^\star \ \text{and} \ \begin{bmatrix} U_0^\star \\ X_0^\star \end{bmatrix}_R)$ are unknown. If the discrepancy between the actual input-output values and measured data is too large, the collected data provide very little insight for any control. The following assumption on the boundedness of the discrepancy is therefore justified in the sense of usefulness of the data.

Assumption 3. (Bounds on the noisy data). There exist δ_X and δ_{UX}

$$X_1^{\star} = X_1 + \delta_X, \qquad \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} = \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} + \delta_{UX} \qquad (12)$$

and

$$\frac{\|\delta_X\|}{\|X_1^*\|} \le r_{X,1}, \quad \frac{\|\delta_{UX}\|}{\|\begin{bmatrix} U_0^* \\ X_0^* \end{bmatrix}\|} \le r_{UX,0} < 1. \tag{13}$$

Assumption 3 only poses bounds on the norm of total noise of the data, so there could be a few outlier data points that can be smoothed out by the rest of accurate data points. We post additional condition of $r_{UX,0} < 1$ for the proof of Theorem III.1, which provides a bound for the estimation error of the system matrix, $\delta_{BA} := \begin{bmatrix} B & A \end{bmatrix} - \begin{bmatrix} B^e & A^e \end{bmatrix}$.

Theorem III.1. (Bound of the estimation error of the system model). If Assumptions 1 and 3 hold and,

$$\operatorname{rank}\left(\begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix}\right) = m + n,\tag{14a}$$

$$\left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\| < \infty, \qquad \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\| < \infty, \tag{14b}$$

then

$$\frac{\|\delta_{BA}\|}{\|[B \quad A]\|} \le c_{UX} \frac{r_{X,1} + r_{UX,0}}{1 - r_{UX,0}},\tag{15}$$

where
$$c_{UX} = \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\| \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^\dagger \right\|$$
 is the condition number.

Proof. The key element of the proof is estimating how the errors of the data propagate to the pseudo inverse of the data matrix $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$ when computing $\begin{bmatrix} B & A \end{bmatrix}$. Defining a matrix

$$\delta_{UX^\dagger}$$
 such that $\begin{bmatrix} U_0^\star \\ X_0^\star \end{bmatrix}_R^\dagger = \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^\dagger + \delta_{UX^\dagger}$, we have

$$\begin{bmatrix}
U_{0}^{\star} \\
X_{0}^{\star}
\end{bmatrix} \begin{bmatrix}
U_{0}^{\star} \\
X_{0}^{\star}
\end{bmatrix}_{R}^{\dagger} = I$$

$$\Rightarrow \left(\begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix} + \delta_{UX}\right) \left(\begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix}_{R}^{\dagger} + \delta_{UX^{\dagger}}\right) = I$$

$$\Rightarrow \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix} \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix}_{R}^{\dagger} + \delta_{UX} \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix}_{R}^{\dagger} + \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix} \delta_{UX^{\dagger}} + \delta_{UX}\delta_{UX^{\dagger}} = I$$

$$\Rightarrow \delta_{UX^{\dagger}} = -\begin{bmatrix}
U_{0}^{\star} \\
X_{0}^{\star}
\end{bmatrix}_{L}^{\dagger} \delta_{UX} \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix}_{R}^{\dagger}$$

$$\Rightarrow \delta_{UX^{\dagger}} = -\begin{bmatrix}
U_{0}^{\star} \\
X_{0}^{\star}
\end{bmatrix}_{R}^{\dagger} \delta_{UX} \begin{bmatrix}
U_{0} \\
X_{0}
\end{bmatrix}_{R}^{\dagger},$$
(16)

where from the third to the fourth line we use the property that $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^\dagger = I$. By definitions of δ_{BA} , $\begin{bmatrix} B & A \end{bmatrix}$, and $\begin{bmatrix} B^e & A^e \end{bmatrix}$, we also have

$$\delta_{BA} = X_1^{\star} \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix}_R^{\dagger} - X_1 \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^{\dagger}$$

$$= (X_1 + \delta_X) \left(\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^{\dagger} + \delta_{UX^{\dagger}} \right) - X_1 \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}_R^{\dagger}$$

$$= \delta_X \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix}_R^{\dagger} + X_1 \delta_{UX^{\dagger}}.$$
(17)

Substituting (16) to (17) gives

$$\delta_{BA} = \delta_{X} \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix}_{R}^{\dagger} (X_{1}^{\star} - \delta_{X}) \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix}_{R}^{\dagger} \delta_{UX} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger}$$

$$= \delta_{X} \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix}_{R}^{\dagger} \begin{bmatrix} B & A \end{bmatrix} \delta_{UX} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} + \delta_{X} \delta_{UX^{\dagger}}$$

$$= \delta_{X} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \begin{bmatrix} B & A \end{bmatrix} \delta_{UX} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger}. \tag{18}$$

Taking the norm on both sides of (18) gives

$$\|\delta_{BA}\| \leq \|\delta_{X}\| \left\| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \right\| + \| \begin{bmatrix} B & A \end{bmatrix} \| \|\delta_{UX}\| \left\| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \right\|$$

$$\Rightarrow \frac{\|\delta_{BA}\|}{\| \begin{bmatrix} B & A \end{bmatrix} \|} \leq \frac{\|\delta_{X}\|}{\| \begin{bmatrix} B & A \end{bmatrix} \|} \left\| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \right\| + \|\delta_{UX}\| \left\| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \right\|$$
Substituting
$$\| \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix} \| \geq \frac{\|X_{1}^{\star}\|}{\| \begin{bmatrix} B & A \end{bmatrix} \|}$$
 to (19), we get
$$\frac{\|\delta_{BA}\|}{\| \begin{bmatrix} B & A \end{bmatrix} \|} \leq \| \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix} \| \| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \| \left(\frac{\|\delta_{X}\|}{\|X_{1}^{\star}\|} + \frac{\|\delta_{UX}\|}{\| \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix} \|} \right)$$

$$= (r_{X,1} + r_{UX,0}) \| \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}_{R}^{\dagger} \| \| \begin{bmatrix} U_{0}^{\star} \\ X_{0}^{\star} \end{bmatrix} \|.$$
 (20)

It is more preferable to have the bound in $\left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\|$ instead of

the unknown
$$\left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\|$$
. Therefore, we work around to bound $\left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\|$ by $\left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\|$,
$$\left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\| \le \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\| + \left\| \delta_{UX} \right\| = \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\| + r_{UX,0} \left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\|$$

$$\implies \left\| \begin{bmatrix} U_0^{\star} \\ X_0^{\star} \end{bmatrix} \right\| \le \frac{1}{1 - r_{UX,0}} \left\| \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right\|. \tag{21}$$

Substituting (21) to (20) completes the proof. \Box

The caveat of the bound Theorem III.1 provides, (15), is that the denominator, $\begin{bmatrix} B & A \end{bmatrix}$, is unknown. By imposing a stronger assumption, Corollary III.2 gives a bound on δ_{BA} that is more practically implementable.

Corollary III.2. (Bound of the estimation error of the system model). If Assumptions 1, 3, Eq. (14) hold, and

$$c_{UX}\frac{r_{X,1} + r_{UX,0}}{1 - r_{UX,0}} := \bar{c} < 1, \tag{22}$$

then

$$\|\delta_{BA}\| \le \frac{\overline{c}}{1-\overline{c}} \| [B^e \quad A^e] \| \tag{23}$$

Proof. The proof is straightforward. By (15), we have

$$\|\delta_{BA}\| \leq \overline{c} \| [B \quad A] \|$$

$$\Longrightarrow \|\delta_{BA}\| \leq \overline{c} \Big(\| [B^e \quad A^e] \| + \|\delta_{BA}\| \Big).$$

Rearranging the terms in the equation above completes the proof. \Box

Note that for $\bar{c} < 1$, $r_{X,1}$ should be strictly less than one because the condition number $c_{UX} \geq 1$. Therefore, although we do not specifically impose an upper bound on $r_{X,1}$, it practically needs to be small enough so that the bound (23) can be derived. In general, there could be some other ways to bound the error. However, to the best of our knowledge, the tightness of the bounds is predominated by the condition number c_{UX} and there is no way to remove the condition number from the bounds. Furthermore, the condition number c_{UX} is determined by the raw data and there is not much control over it. Though there is some research on effective condition number for positive definite matrices that can potentially tighten the bounds [18], [19], how the concept extends to common rectangular matrices is unclear. We next discuss two routes to pre-conditioning the data matrix.

B. Pre-conditioning the data matrix

There are two benefits of pre-conditioning the data matrix by reducing its condition number. One is a (numerically) tighter bound for robust controller design. The other is that the actual modeling error propagated from the noisy data can be reduced. The pre-conditioning is non-trivial, so we only go through some potentially useful methods that reduce the condition number.

The first way to pre-condition the data matrix is by appropriate selection of data points. Recall that the only requirement for the data representation of the original system model is the full row rank of $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$ and we construct U_0 (or X_0) by using all the data from $k=0,\cdots,T$. The idea is that by selecting a subset of the T+1 data points to construct U_0 (or X_0), the condition number can become smaller, while the full row rank condition still holds. [20] provides some useful insights in this route, particularly the following theorem:

Theorem III.3. (Bougain-Tzafriri [20]). Suppose matrix A is standardized. Then there is a set τ of column indices for which

$$|\tau| \ge c \cdot \frac{\|A\|_F}{\|A\|}$$

such that the sub-matrix of A indexed by τ has the condition number less than or equal to $\sqrt{3}$.

The constant c in Theorem III.3 refers to a positive, universal constant. The upper bound of $\sqrt{3}$ is very decent such that the bound in Theorem III.1 is tight in the sense that there is limited amplification of data errors toward the modeling error. What makes Theorem III.1 more compelling is that there is an algorithmic version available [20, Algorithm 2]. However, practically, we may not find a feasible selection of the columns that gives the bound close to $\sqrt{3}$. One of the main reasons is that Theorem III.3 (or [20, Algorithm 2]) counts the option of non-full row rank selection of columns, or vertical matrices such that Assumption 1 does not hold. Nevertheless, one can

modify [20, Algorithm 2] with an additional constraint on the number of columns in an attempt to improve the condition number by not using the full data set.

Another way to improve the condition number is diagonal scaling. The goal is finding diagonal matrices, D_L and D_R , such that the condition number of $\overbrace{\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}} := D_L \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} D_R$ is smaller than $\begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$. This diagonal rescaling does not change

the structure of the linear equality that we aim to solve:

$$X_{1} = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}$$

$$\iff X_{1}D_{R} = \begin{bmatrix} B & A \end{bmatrix} D_{L}^{-1} \left(D_{L} \begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix} D_{R} \right)$$

$$\iff \widehat{X}_{1} = \widehat{\begin{bmatrix} B & A \end{bmatrix}} \widehat{\begin{bmatrix} U_{0} \\ X_{0} \end{bmatrix}}, \tag{24}$$

where we define $\widehat{X}_1 = X_1 D_R$ and $\widehat{BA} = \widehat{BA} D_L^{-1}$. By repeating the steps in section III-A for (24) instead of (3), we can get the bound of the error term $\delta_{\widehat{BA}}$ relative to \widehat{BA} as in (15) in Theorem III.1, which is tighter than the original one in the sense that the condition number for \widehat{A} is smaller than \widehat{A} . Note that we can not conclude analytically that

the resulting $\begin{bmatrix} B & A \end{bmatrix}$ deduced from $\begin{bmatrix} \widehat{B} & A \end{bmatrix}$ has a smaller modeling error (originated from noisy data) compared to the case without the diagonal scaling. However, numerical studies imply that diagonal scaling could reduce the errors. More details about heuristic algorithms for the diagonal scaling can be found in [21], [22].

C. Robust controller design for LTI systems

In this subsection, we leverage the modeling bounds in section III-A for a robust controller design for LTI systems. Our goal here is designing a feedback gain $K \in \mathbb{R}^{m \times n}$ such that (1) is stable without knowing A and B matrices but some noisy input-output data that construct U_0 , X_0 , and X_1 . We assume that there is an upper bound on the feedback gain, denoted as $\|K\| \leq \overline{K}$. Theorem III.4 shows a formulation that finds the robust feedback gain, K.

Theorem III.4. (Data-driven robust controller for LTIs). If Assumption 1, 3 holds and $\exists \lambda \in \mathbb{R}_+$, $Q \in \mathbb{R}^{T \times n}$, $\Delta X_1 \in \mathbb{R}^{n \times T}$ such that

$$X_0 Q = I, (25a)$$

$$\begin{bmatrix} \overline{K}I & U_0Q\\ (U_0Q)^\top & \overline{K}I \end{bmatrix} \succeq 0, \tag{25b}$$

$$\begin{bmatrix} (1 - \lambda \overline{\Delta X_1})I & X_1Q + (1 - \lambda)\Delta X_1Q \\ Q^{\top}X_1^{\top} + (1 - \lambda)Q^{\top}\Delta X_1^{\top} & (1 - \lambda \overline{\Delta X_1})I \end{bmatrix} \succeq 0,$$
(25c)

where
$$\overline{\Delta X_1} = \frac{\overline{c}}{1-\overline{c}} \| [B^e \quad A^e] \| \| \overline{K}_I \|$$
. Then $u = Kx$, $K := U_0 Q(X_0 Q)^{-1}$, stabilizes system (1).

Proof. A stability condition for a feedback control u = Kx is given by

$$\exists P \succeq 0 \text{ s.t. } (A + BK)P(A + BK)^{\top} \preceq P.$$
 (26)

Finding a K that stabilizes the linear system is all about solving linear matrix inequalities shown in (26) when A and B are known. Similar to the method in [7], we define

$$\begin{bmatrix} K \\ I \end{bmatrix} = \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} G, \tag{27}$$

which leads to

$$A + BK = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} K \\ I \end{bmatrix} = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} G \qquad (28)$$
$$= \left(\begin{bmatrix} B^e A^e \end{bmatrix} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} + \delta_{BA} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} \right) G$$

By defining Q = GP and $\Delta X_1 = \delta_{BA} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix}$, we rewrite (26) in the following

$$\exists P \succeq 0 \text{ s.t. } (X_1 + \Delta X_1)GP(P)^{-1}PG^{\top}(X_1 + \Delta X_1)^{\top} \preceq P$$

$$\Leftrightarrow \exists P \succeq 0 \text{ s.t. } (X_1 + \Delta X_1)Q(P)^{-1}Q^{\top}(X_1 + \Delta X_1)^{\top} \preceq P$$

$$\Leftrightarrow \begin{bmatrix} X_0Q & (X_1 + \Delta X_1)Q \\ Q^{\top}(X_1 + \Delta X_1)^{\top} & X_0Q \end{bmatrix} \succeq 0, \tag{29}$$

$$X_0Q = P \succeq 0,$$

where we apply Schur complement and use (27) in the derivations above. Note that the decision variable is changed from K and P to Q as both K and P can be uniquely derived from Q. Specifically, $K = U_0G = U_0QP^{-1} = U_0Q(X_0Q)^{-1}$. The formulation (29) requires knowledge of ΔX_1 which we do not have in general. An alternative and robust reformulation is to require (29) holds for all possible ΔX_1Q , where the results in Theorem III.1 or Corollary III.2 become handy. Without loss of generality, we let P = I, and then write the upper bound of K and ΔX_1Q as shown in the following:

$$||K|| = ||U_0 Q|| \le \overline{K}$$

$$||\Delta X_1 Q|| = \left\| \delta_{BA} \begin{bmatrix} U_0 \\ X_0 \end{bmatrix} Q \right\| \le ||\delta_{BA}|| \left\| \begin{bmatrix} K \\ I \end{bmatrix} \right\| \le \overline{\Delta X_1}.$$
(30a)

Rewriting (30) to LMIs gives (25b) and

$$\begin{bmatrix} \overline{\Delta X_1} I & \Delta X_1 Q \\ (\Delta X_1 Q)^\top & \overline{\Delta X_1} I \end{bmatrix} \succeq 0.$$
 (31)

An alternative robust reformulation of (29) is to require that (29) holds for all ΔX_1 satisfying (31) with $X_0Q = P = I$. By applying S-procedure, we derive (25c) and complete the proof.

The matrix inequalities condition in Theorem III.4 are non-linear because of the bilinear term $(1-\lambda)\Delta X_1Q$, which pose some challenge to solve. An alternative convex formulation of (25) is available. Because ΔX_1 does not show up in any other part of (25) other than (25c), we can define a new variable $\Delta Q := \Delta X_1Q$ to bypass the bilinear term ΔX_1Q

and retrieve ΔX_1 afterward. The only remaining bilinear term is on $\lambda \Delta Q$. Numerically, we can fix λ at a small value and check the feasibility of (25). If (25) is feasible for the given λ , we can still find a robust feedback control gain, K.

D. Robust controller design for switched linear systems

We extend the robust data-driven control to the switched linear system. The objective is to derive feedback controls $u = K_i x$ for $i \in \Gamma$ such that the switched linear system is stable under random switching. Finding a common Lyapunov function in the following guarantees the stability

$$\exists P \succ 0 \text{ s.t. } (A_i + B_i K_i) P(A_i + B_i K_i)^{\top} P, \forall i \in \Gamma.$$
 (32)

Applying Lemma II.1, we can identify the system matrices for each mode $i \in \Gamma$ with data of randomly switching sequence. Similar to the last subsection, for each $i \in \Gamma$, we define G_i by

$$\begin{bmatrix} K_i \\ I \end{bmatrix} = \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} G_i,$$

which leads to

$$A_{i} + B_{i}K_{i} = [B_{i} \ A_{i}] \begin{bmatrix} K_{i} \\ I \end{bmatrix} = [B_{i} \ A_{i}] \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} G_{i}$$

$$= \left([B_{i}^{e} \ A_{i}^{e}] + \delta_{BA_{i}} \right) \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} G_{i}$$

$$= \left(\left(X_{1} - \sum_{j \in \Gamma, j \neq i} [B_{j}^{e} \ A_{j}^{e}] \begin{bmatrix} U_{j,0} \\ X_{j,0} \end{bmatrix} \right) + \delta_{BA_{i}} \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} \right) G_{i}$$

$$= \left(\left(X_{1} - \sum_{j \in \Gamma, j \neq i} [X_{j,0}^{U_{j,0}}]^{\dagger} \begin{bmatrix} U_{j,0} \\ X_{j,0} \end{bmatrix} \right) + \delta_{BA_{i}} \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} \right) G_{i}$$

$$= \left(X_{1} \left(I - \sum_{j \in \Gamma, j \neq i} [U_{j,0} \\ X_{j,0} \end{bmatrix}^{\dagger} \begin{bmatrix} U_{j,0} \\ X_{j,0} \end{bmatrix} \right) + \delta_{BA_{i}} \begin{bmatrix} U_{i,0} \\ X_{i,0} \end{bmatrix} \right) G_{i} .$$

Equation (33) is analogous to (28) with the only difference on an additional term that multiplies with X_1 dependent on the mode $i \in \Gamma$. From this point on, we can introduce Q_i for each $i \in \Gamma$ repeat the steps in the previous subsection to formulate a semidefinite programming similar to (25) for the robust control feedback gains. We skip those similar derivations that involve tedious notations with only marginal additional insights.

IV. NUMERICAL RESULTS

We consider a switched system in the form of (6) with $x \in \mathbb{R}^{20}, u \in \mathbb{R}^{10}$ and $|\Gamma| = 5$. All the elements of B_i for each $i \in \Gamma$ are randomly generated between 0 and 0.1; A_i for each $i \in \Gamma$ is generated as $A_i = 0.9 \cdot I + \Delta A_i$, where all the elements of ΔA_i are also randomly generated between 0 and 0.1. For the first 500 steps, we run the system under random control and switching for the purpose of collecting data. The measurement noise of the state x is uniformly distributed between -0.5% and 0.5% of the absolute value of x. We first apply the results in Section II-B to identify $[B_i^e \ A_i^e]$ for each $i \in \Gamma$. We use the bound in Corollary III.2 for the purpose of robust controller design. The bound, however, is too conservative. Therefore,

we applied the pre-conditioning methods including column selection using [20, Algorithm 2] (with an additional lower bound of the number of columns) and diagonal scaling using Ruiz algorithm in [21]. The column selection method was found not improving the condition number for a tighter bound. Our explanation for the ineffectiveness of the column selection are (i) the additional lower bound on the number of columns; (ii) [20, Algorithm 2] is randomized by nature and may only useful for certain classes of matrices. The diagonal scaling, on the other hand, improve the condition numbers of all the mode by a factor of around 10 as shown in Table I. The improved condition numbers are directly reflected on tighter bounds of the estimation errors as shown in Table II. In addition, the actual estimation errors are reduced marginally with the diagonal scaling, shown in Table III. Those far better condition numbers benefit the robust controller design. We next apply the results in Section III to find robust feedback control gains K_i for all $i \in \Gamma$. Figure 1 shows the trajectory of x under the robust feedback control. As expected, x converges to the origin within a moderate number of steps under the control.

	w/o pre-conditioning	w pre-conditioning
Mode 1	199.1373	21.0689
Mode 2	136.7279	16.3103
Mode 3	160.5263	18.2697
Mode 4	173.2082	18.6434
Mode 5	170.2047	20.3172

The condition number of $[U_{i,0}^{\top}X_{1,0}^{\top}]^{\top}$ w/wo pre-conditioning the data matrices.

	w/o pre-conditioning	w pre-conditioning		
Mode 1	4.0230	0.4256		
Mode 2	2.7622	0.3295		
Mode 3	3.2430	0.3691		
Mode 4	3.4992	0.3766		
Mode 5	3.4385	0.4104		
TADIEII				

THE UPPER BOUNDS OF $\frac{\|\delta_{BA}\|}{\|[BA]\|}$ W/WO PRE-CONDITIONING THE DATA MATRICES

w/o pre-conditioning	w pre-conditioning	% changes
0.0136	0.0115	15%
0.0115	0.0095	17%
0.0130	0.0125	4%
0.0165	0.0155	6%
0.0129	0.0125	3%
	0.0136 0.0115 0.0130 0.0165	0.0136 0.0115 0.0115 0.0095 0.0130 0.0125 0.0165 0.0155

The value of $\frac{\|\delta_{BA}\|}{\|[BA]\|}$ W/wo pre-conditioning the data matrices.

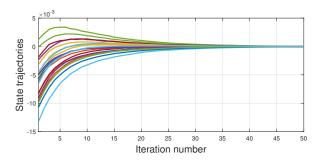


Fig. 1. State trajectories under the proposed data-driven robust controller.

V. CONCLUSION

In this paper, we analyze system identification errors originated from noisy data and methods of pre-conditioning the data to improve the error bounds. The bounds on the inaccurate modeling are incorporated in robust controller design for LTI systems and switched linear systems. In the future, we will migrate the focus toward real-world applications and make necessary adjustments depending on the application needs.

REFERENCES

- A. Alanwar, Y. Stürz, and K. Johansson, "Robust data-driven predictive control using reachability analysis," *European Journal of Control*, vol. 68, 2022.
- [2] J. Willems, "System theoretic models for the analysis of physical systems," *Ricerche di Automatica*, vol. 10, 1979.
- [3] J. C. Willems, "From time series to linear system Part I: Finite dimensional linear time invariant systems," *Automatica*, vol. 22, no. 5, pp. 561–580, 1986.
- [4] _____, "From time series to linear system Part II: Exact modelling," *Automatica*, vol. 22, no. 6, pp. 675–694, 1986.
- [5] —, "From time series to linear system Part III: Approximate modelling," *Automatica*, vol. 23, no. 1, pp. 87–115, 1987.
- [6] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," in 2019 18th European Control Conference (ECC). IEEE, 2019, pp. 307–312.
- [7] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 909–924, 2019.
- [8] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2020.
- [9] J. Berberich, A. Koch, C. W. Scherer, and F. Allgöwer, "Robust datadriven state-feedback design," in *American Control Conference (ACC)*, 2020, pp. 1532–1538.
- [10] C. De Persis and P. Tesi, "Low-complexity learning of linear quadratic regulators from noisy data," *Automatica*, vol. 128, p. 109548, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/ S0005109821000686
- [11] C. Verhoek, R. Tóth, S. Haesaert, and A. Koch, "Fundamental lemma for data-driven analysis of linear parameter-varying systems," arXiv preprint arXiv:2103.16171, 2021.
- [12] M. Rotulo, C. D. Persis, and P. Tesi, "Online learning of data-driven controllers for unknown switched linear systems," 2021.
- [13] S. Baros, C.-Y. Chang, G. E. Colon-Reyes, and A. Bernstein, "Online data-enabled predictive control," to appear in Automatica, 2020.
- [14] H. J. Van Waarde, J. Eising, H. L. Trentelman, and M. K. Camlibel, "Data informativity: a new perspective on data-driven analysis and control," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4753–4768, 2020.
- [15] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (gpca)," *IEEE transactions on pattern analysis and machine* intelligence, vol. 27, no. 12, pp. 1945–1959, 2005.
- [16] T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Readings in computer vision*, pp. 638–643, 1987.
- [17] H. Song, M. Kim, D. Park, Y. Shin, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
 [18] T. F. Chan and D. E. Foulser, "Effectively well-conditioned linear
- [18] T. F. Chan and D. E. Foulser, "Effectively well-conditioned linear systems," SIAM journal on scientific and statistical computing, vol. 9, no. 6, pp. 963–969, 1988.
- [19] Z.-C. Li, C.-S. Chien, and H.-T. Huang, "Effective condition number for finite difference method," *Journal of computational and applied* mathematics, vol. 198, no. 1, pp. 208–235, 2007.
- [20] J. A. Tropp, "Column subset selection, matrix factorization, and eigenvalue optimization," in *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2009, pp. 978–986.
- [21] R. Takapoui and H. Javadi, "Preconditioning via diagonal scaling," arXiv preprint arXiv:1610.03871, 2016.
- [22] A. M. Bradley, "Algorithms for the equilibration of matrices and their application to limited-memory quasi-newton methods," STANFORD UNIV CA, Tech. Rep., 2010.