Contents lists available at ScienceDirect







journal homepage: www.elsevier.com/locate/arcontrol

Closed-form and robust expressions for data-driven LQ control*

Federico Celi^b, Giacomo Baggio^a, Fabio Pasqualetti^{b,*}

^a Department of Information Engineering, University of Padova, Padova, 35131, Italy

^b Department of Mechanical Engineering, University of California Riverside, Riverside, CA 92521, USA

ARTICLE INFO

Keywords: Data-driven control Optimal control Robustness

ABSTRACT

This article provides an overview of certain direct data-driven control results, where control sequences are computed from (noisy) data collected during offline control experiments without an explicit identification of the system dynamics. For the case of noiseless datasets, we derive several closed-form data-driven expressions that solve a variety of optimal control problems for linear systems with quadratic cost functions of the state and input (including the linear quadratic regulator problem, the minimum energy control problem, and the linear quadratic control problem with terminal constraints), discuss their advantages and drawbacks with respect to alternative data-driven and model-based approaches, and showcase their effectiveness through a number of numerical studies. Interestingly, these results provide an alternative and explicit way of solving classic control problems sthat, for instance, does not require the solution of an implicit and recursive Riccati equation as in the noiseless setting. For the case of noisy datasets, we show how the closed-form expressions derived in the noiseless setting can be modified to compensate for the bias induced by noise, and perform a sensitivity analysis to reveal favorable asymptotic robustness properties of the derived data-driven controls. We conclude the paper with some considerations and a discussion of outstanding questions and directions of future investigation.

1. Introduction

Data-driven control refers to the design of algorithms for systems with unknown dynamics using data obtained from a set of control experiments. This approach does not require the modeling of the system dynamics using first principles, which is instead needed by the traditional model-based methods and can be challenging for certain complex systems. The design of data-driven controls can be indirect, where the control sequence is computed using a model of the system dynamics identified from data (i.e., the identification plus control pipeline), or direct, where the control sequence is computed in an end-to-end fashion without explicitly estimating the system dynamics from data. While model-based and indirect data-driven control design have a long and widely accepted history, direct data-driven control has received renewed attention, possibly motivated by the numerous successes of machine learning and artificial intelligence techniques.

This article reviews some of the recent results for direct datacontrol of linear time-invariant systems. Our main premise is that, especially for systems with linear time-invariant dynamics, systemtheoretic properties and control methods are well-understood and have been developed over the years, ranging from tests and algorithms in the frequency domain Åström and Murray (2010) to methods using statespace (Kailath, 1980) and geometric computations (Basile & Marro,

1991), among others. The analysis, design and synthesis of control methods based on the system behaviors (Willems & Polderman, 1997) is certainly possible and interesting, but perhaps most useful only if complemented with an understanding of when a specific problem should be solved within a specific domain (e.g., frequency, state space, behaviors) and using a specific algorithm. This type of questions, which has been present, for instance in the machine learning research to characterize the tradeoffs between generative and discriminative models (Ng & Jordan, 2001), has instead received only scarce attention in the context of data-driven control. As we shall see later, while data-driven and modelbased methods are theoretically equivalent in the absence of noise and assuming perfect computations, in practice, the methods can differ considerably in the way they propagate uncertainties thus leading unexpectedly to different results. Furthermore, even within the data-driven framework, different formalisms can lead to expressions with different complexity, interpretability and performance, thus motivating careful analysis and comparisons. We will focus on a specific (and, in our opinion, simple and insightful) framework to solve Linear Quadratic (LQ) control problems using data, and comment on the advantages and disadvantages of the proposed solutions. We refer the interested reader to Markovsky and Dörfler (2021) for a more comprehensive survey of data-driven control methods and the behavioral framework.

https://doi.org/10.1016/j.arcontrol.2023.100916

Received 20 July 2023; Accepted 1 November 2023

[☆] This material is based upon work supported in part by AFOSR, USA awards FA9550-20-1-0140, and FA9550-19-1-0235. * Corresponding author.

E-mail addresses: fceli001@ucr.edu (F. Celi), baggio@dei.unipd.it (G. Baggio), fabiopas@ucr.edu (F. Pasqualetti).

^{1367-5788/© 2023} The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

The rest of the paper is organized as follows. Section 2 introduces the problem setting, our notation, and some preliminary results on representing linear dynamical systems using data. In particular, Lemma 2.1 contains a reformulation of Willems' Fundamental Lemma (Willems, Rapisarda, Markovsky, & De Moor, 2005) that allows for the explicit characterization of the free and forced responses of a linear system using data, and extends the state-space data-based representation in van Waarde, De Persis, Camlibel and Tesi (2020). Section 3 contains explicit data-driven formulas for the solution to a variety of LQ control problems. These results extend and generalize the ones obtained in Baggio, Katewa, and Pasqualetti (2019) and Baggio, Bassett, and Pasqualetti (2021) for the case of a single dataset, and the ones in Baggio and Pasqualetti (2020) and van Waarde, De Persis et al. (2020). Lemma 3.3, in particular, further extends the data-based representation of the system dynamics in Lemma 2.1 to allow for the use of heterogeneous datasets and for the representation of trajectories longer than the ones available in the dataset. Section 4 contains our formulas for data-driven LQ control with noisy datasets, and contains (i) closed-form expressions that compensate for the effect of noise (when the noise statistics are known) and are asymptotically consistent, and (ii) a sensitivity study that shows that the effect of perturbations vanish asymptotically, when their magnitude is bounded and they affect only a "sublinear" number of datapoints. These results generalize our previous bounds in Baggio et al. (2021) and Celi, Baggio, and Pasqualetti (2023b). The case of closed-loop LQ control is treated in Section 5, where we follow the ideas in Celi, Baggio, and Pasqualetti (2022) and show how the Linear Quadratic Regulator (LQR) gains can be learned in a data-driven setting without identifying the system dynamics nor solving iterative, implicit Riccati equations (Theorem 5.1 contains our data-driven closed-form expressions of the LQR gain). Finally, Section 6 concludes the paper.

Notation. We let \mathbb{R} and \mathbb{N} denote the set of real and integer numbers, respectively. Given a matrix $A \in \mathbb{R}^{n \times m}$, we let $\operatorname{Rank}(A)$, $\operatorname{Basis}(A)$, $\operatorname{Ker}(A)$, A^{T} , $\sigma_{\min}(A)$, $\sigma_{\max}(A)$ denote the rank, a basis of the column space, the kernel, the transpose, and the smallest and largest singular values of A, respectively. We let blk-diag (A_1, \ldots, A_n) be the block diagonal matrix with blocks $A_i \in \mathbb{R}^{n_i \times m_i}$. We denote the Moore–Penrose pseudoinverse of matrix A with A^{\dagger} . We indicate the 2-norm of a matrix or vector with $\|\cdot\|_2$. We let A > 0 ($A \ge 0$) denote a positive definite (positive semidefinite) matrix. We let vec(A) be the vectorization of matrix A. For a positive semidefinite matrix $W \in \mathbb{R}^{n \times n}$ and vector $x \in \mathbb{R}^n$, we let $||x||_W = \sqrt{x^T W x}$. I_n and 0_{nm} denote the $n \times n$ identity matrix and $n \times m$ zero matrix, respectively (subscripts will be omitted when clear from the context). For a random vector $x : \Omega \to \mathbb{R}^n$, we let $\mathbb{P}[x \in S]$ and $\mathbb{E}[x]$ be the probability that *x* takes on a value in a set $S \subseteq \mathbb{R}^n$ and the expected value of x, respectively. We let a.s. denote almost surely, and $\xrightarrow{a.s.}$ almost sure convergence.

2. Problem setting and preliminary notions

We study the problem of designing control inputs for linear timeinvariant systems to solve a variety of optimal control and robustness problems. We do this without knowing the system dynamics and by, instead, leveraging a set of pre-recorded input–output trajectories, together with state trajectories when needed. In particular, we consider systems with linear, discrete-time, time-invariant dynamics of the form

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $y \in \mathbb{R}^p$ are the state, input and output vectors at time $t \in \mathbb{N}$, respectively, and the matrices *A*, *B*, *C* and *D* are unknown. We assume that inputs *U*, states *X*, and outputs *Y* are available from a set of $N \in \mathbb{N}$ control experiments with finite horizon $T \in \mathbb{N}$:

$$X_0 = \begin{bmatrix} x^1(0) & \cdots & x^N(0) \end{bmatrix} \in \mathbb{R}^{n \times N},$$
(2b)

$$X = \begin{bmatrix} \mathbf{x}_T^1 & \cdots & \mathbf{x}_T^N \end{bmatrix} \in \mathbb{R}^{nT \times N},\tag{2c}$$

$$Y = \begin{bmatrix} \mathbf{y}_T^1 & \cdots & \mathbf{y}_T^N \end{bmatrix} \in \mathbb{R}^{pT \times N}.$$
(2d)

In (2), \mathbf{u}_T^i , \mathbf{x}_T^i , and \mathbf{y}_T^i are the vectors containing the inputs, states, and outputs of the *i*th experiment:

$\mathbf{u}_T^i = \operatorname{vec}(u^i(0), \dots, u^i(T-1)),$	(<i>ith input trajectory</i>)
$\mathbf{x}_T^i = \operatorname{vec}(x^i(1), x^i(2), \dots, x^i(T)),$	(ith state trajectory)
$\mathbf{v}_{T}^{i} = \operatorname{vec}(v^{i}(0), \dots, v^{i}(T-1)).$	(<i>ith</i> output trajectory)

We remark that the full input-state-output dataset (2) is not always needed to solve the problems described in this tutorial paper; the subset of required data will be specified based on the problem at hand. Further, when convenient, we will use of the following matrices, which can be extracted from (2):

$$X_{\rm F} = \begin{bmatrix} x^1(T) & \cdots & x^N(T) \end{bmatrix} \in \mathbb{R}^{n \times N},\tag{3a}$$

$$Y_{\rm F} = \begin{bmatrix} y^1(T-1) & \cdots & y^N(T-1) \end{bmatrix} \in \mathbb{R}^{p \times N}.$$
(3b)

For the noiseless system (1), knowledge of the system matrices *A*, *B*, *C* and *D* is equivalent to the availability of a (sufficiently rich) dataset (2). In fact, any dataset (2) can be generated with the matrices *A*, *B*, *C* and *D* using (1), and, in turn, such matrices can be reconstructed uniquely (under mild data rank conditions) using the dataset (2):¹

$$X_{\rm m}^+ = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} X_{\rm m}^- \\ U_{\rm m} \end{bmatrix}$$
, and $Y_{\rm m} = \begin{bmatrix} C & D \end{bmatrix} \begin{bmatrix} X_{\rm m}^- \\ U_{\rm m} \end{bmatrix}$,

where

$$\begin{split} &U_{\rm m} = \left[u^1(0), \dots, u^1(T-1), \dots, u^N(0), \dots, u^N(T-1) \right], \\ &X_{\rm m} = \left[x^1(0), \dots, x^1(T), \dots, x^N(0), \dots, x^N(T) \right], \\ &X_{\rm m}^- = \left[x^1(0), \dots, x^1(T-1), \dots, x^N(0), \dots, x^N(T-1) \right], \\ &X_{\rm m}^+ = \left[x^1(1), \dots, x^1(T), \dots, x^N(1), \dots, x^N(T) \right], \\ &Y_{\rm m} = \left[y^1(0), \dots, y^1(T), \dots, y^N(0), \dots, y^N(T-1) \right]. \end{split}$$

Clearly, the system matrices can be computed uniquely whenever the matrix $\begin{bmatrix} X_m^-\\ U_m \end{bmatrix}$ is full row rank. While this analysis seems to discourage the pursuit of data-driven methods, since sufficiently-rich datasets are effectively a model of the system dynamics, in this review we will show that, instead, data-driven computations allow for alternative and sometimes more insightful, direct, and computationally-favorable solutions to classic control problems, thus contributing to the theory of systems and enriching our control tools. We remark that a detailed analysis of the tradeoffs between direct data-driven methods and classic identification-based approaches in noisy settings is much more nuanced (Krishnan & Pasqualetti, 2021), deserves a dedicated treatment, and will not be addressed here. Rather, we will show how our direct solutions obtained with noiseless data can be used to study, and modified to counteract, the effect of noise and perturbations on the datasets.

A powerful result underlying most data-driven approaches is Willems' Fundamental Lemma (Willems et al., 2005), which, loosely speaking, gives a sufficient conditions under which any *T*-long trajectory of the system (1) can be constructed as linear combinations of those in an appropriately constructed input–output dataset (Markovsky & Dörfler, 2021). We next state a reformulation of Willems' Fundamental Lemma that uses the dataset (2) and that allows us to distinguish between the trajectories of (1) obtained with and without a control input. This result will be instrumental for our derivations.

¹ Similar formulas are contained also in De Persis and Tesi (2020), among others.

Lemma 2.1 (Data-based Free and Forced Representation Celi & Pasqualetti, 2022). Let (2)–(3) be the data generated by the system (1). Assume that

$$\operatorname{Rank}\left(\begin{bmatrix} X_0\\ U \end{bmatrix}\right) = mT + n. \tag{4}$$

Let K_U = Basis(Ker(U)) and K_0 = Basis(Ker(X_0)). Then, for any initial state x_0 and input \mathbf{u}_T , there exist vectors α and β such that $x_0 = X_0 K_U \alpha$ and $\mathbf{u}_T = U K_0 \beta$. Moreover,

$$\begin{bmatrix} \mathbf{x}_T \\ \mathbf{y}_T \end{bmatrix} = \begin{bmatrix} XK_U & XK_0 \\ YK_U & YK_0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$
(5)

are the state and output trajectories of length T of (1) generated by x_0 and \mathbf{u}_T .

Lemma 2.1 states that any trajectory of the system (1) can be written as a linear combination of a collection of previously-recorded trajectories. Yet, Lemma 2.1 provides a more granular decomposition of the trajectories of the system (1) given data, as it identifies the *free response* of the system from the initial condition x_0 , namely $X K_U \alpha$, and the *forced response* of the system from the input \mathbf{u}_T , namely $X K_0 \beta$. In addition to being of general interest, these formulas allow the recorded data to be used to predict the system trajectories given the initial condition and input sequence, rather than just as a description of the system dynamics, and to analyze certain system-theoretic properties in a purely data-driven manner without requiring the identification of the system dynamics (see also Celi and Pasqualetti (2022)).

Lemma 2.2 (Data-driven Reachability and Observability). Let the data matrices X_0 and U satisfy (4). Then

$$\operatorname{Rank}(O_T^Y) = \operatorname{Rank}(YK_U),\tag{6}$$

where $O_T^Y = \begin{bmatrix} C^T & (CA)^T & \cdots & (CA^{T-1})^T \end{bmatrix}^T$, is the *T*-steps observability matrix of (1). Similarly,

$$\operatorname{Rank}(C_T) = \operatorname{Rank}(X_F K_0),\tag{7}$$

where $C_T = \begin{bmatrix} A^{T-1}B & \cdots & AB & B \end{bmatrix}$ is the *T*-steps controllability matrix of (1) and X_F is as in (3).

Lemma 2.2 relates the rank of the *T*-step observability matrix O_T^Y to the data-driven matrix YK_U . Clearly, when $T \ge n$, the system (1) is observable if and only if $\operatorname{Rank}(YK_U) = n$. Similar comments hold in (7) for the reachability subspace of (1).

Remark 1 (*Single vs. Multiple Data Trajectories*). A single experimental trajectory may be sufficient to obtain a data-driven representation of the system dynamics. A single trajectory, in fact, is used in Willems' Fundamental Lemma and in several reformulations of this result, e.g., Lopez and Müller (2022), Schmitz, Faulwasser, and Worthmann (2022), Verhoek, Tóth, Haesaert, and Koch (2021), Willems et al. (2005) and Yu et al. (2021). We remark that the use of multiple trajectories, as we do in Lemma 2.1, carries some advantages. First, the formulas in Lemma 2.1 remain valid if only a single, long, trajectory is available. In fact, the case of a single, long trajectory organized as a Hankel matrix is a special case of our formalism. To see this, using the notation in van Waarde, De Persis et al. (2020), the data collected from a single trajectory of length τ is organized as

$$\begin{aligned} \mathcal{H}_1(x) &= \begin{bmatrix} x(0) & x(1) & \cdots & x(\tau - T) \end{bmatrix}, \\ \mathcal{H}_T(u) &= \begin{bmatrix} u(0) & u(1) & \cdots & u(\tau - T) \\ \vdots & \vdots & & \vdots \\ u(T - 1) & u(T) & \cdots & u(\tau - 1) \end{bmatrix}. \end{aligned}$$

Clearly, one can set $X_0 = \mathcal{H}_1(x)$ and \mathbf{u}_T^i equal to the *i*th column of $\mathcal{H}_T(u)$ to equivalently express the data as in our framework. Thus, considering multiple, short trajectories as in (2) effectively generalizes and includes the case of a single, long trajectory, Second, the use of

multiple trajectories is convenient when the dynamics are unstable, since the system needs to be simulated for a shorter time horizon compared to the case of a single trajectory. More generally, using multiple trajectories produces data matrices that are numerically better conditioned and yield more reliable computations. Finally, the use of multiple trajectories is convenient from a statistical perspective when the collected data is corrupted by noise (Tu, Frostig, & Soltanolkotabi, 2022). \Box

Remark 2 (*State vs. Output Measurements*). We assume here that the state of the system (1) can be directly measured. Notice, however, that this is not a restrictive assumption since a state measurement can be replaced with a finite window of inputs and outputs to solve appropriate control problems. In fact, the dynamics of (1) can be equivalently written using only inputs and outputs as done, e.g., in Al Makdah, Krishnan, Katewa, and Pasqualetti (2022) for the data-driven LQG control problem. \Box

3. Data-driven formulas for open-loop LQ control

We start by studying the LQ control problem

$$\begin{array}{ll} \underset{u, x, y}{\text{minimize}} & \sum_{t=0}^{T-1} \left(\|y(t)\|_{Q_t}^2 + \|u(t)\|_{R_t}^2 \right) \\ \text{subject to} & x(t+1) = Ax(t) + Bu(t), \\ & y(t) = Cx(t) + Du(t), \end{array}$$
(8)

$$x(0) = x_0, \ y(T-1) = y_f,$$

where $Q_t \geq 0$ and $R_t > 0$ are the (time-varying) output and input weighting matrices, x_0 is the initial state, and y_f the desired value of the output at time T - 1. Throughout the paper, we assume that (1) is output controllable in T - 1 steps² to guarantee feasibility of (8) for any choice of $x_0 \in \mathbb{R}^n$ and $y_f \in \mathbb{R}^p$, and that the rank condition (4) holds. Problem (8) generalizes the classic (open-loop) linear–quadratic control framework by including the possibility of minimizing a linear function of the state (as opposed to the whole state) in addition to the control input. Let $\mathbf{u}_T = \operatorname{vec}(u(0), \ldots, u(T - 1))$ be the vector of inputs, and

$$Q = \text{blk-diag}(Q_0, \dots, Q_{T-1}),$$

$$R = \text{blk-diag}(R_0, \dots, R_{T-1}).$$
(9)

Using the notation in Section 2, we now present a closed-form solution to the LQ control problem (8) that relies only on the data collected in (2). First, notice that the cost in (8) can be written in vector form as

$$\sum_{t=0}^{T-1} \left(\left\| y(t) \right\|_{Q_t}^2 + \left\| u(t) \right\|_{R_t}^2 \right) = \mathbf{y}_T^\mathsf{T} Q \mathbf{y}_T + \mathbf{u}_T^\mathsf{T} R \mathbf{u}_T$$

Second, from Lemma 2.1 we have $\mathbf{y}_T = Y K_U \alpha + Y K_0 \beta$ and $\mathbf{u}_T = U K_0 \beta$, for some vectors α and β . Then, by letting $K = \begin{bmatrix} K_U & K_0 \end{bmatrix}$, the input and output trajectories can be equivalently written as

$$\mathbf{y}_T = YK \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$
 and $\mathbf{u}_T = UK \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$,

respectively, and the cost in (8) becomes

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix}^{\mathsf{T}} \left((YK)^{\mathsf{T}} Q(YK) + (UK)^{\mathsf{T}} R(UK) \right) \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

Similarly, the equality constraints in (8) can be written as

 $\begin{bmatrix} X_0 \\ Y_F \end{bmatrix} K \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} x_0 \\ y_f \end{bmatrix}.$

² System (1) is output controllable in T steps if $[CA^{T-1}B \cdots CAB CB D]$ has full row rank. We remark that Lemma 2.2 can be adapted to verify whether a system is output controllable directly from data, substituting X_F with Y_F .

The above reasoning allows us to reformulate the LQ control problem (8) as the data-based problem

$$\begin{array}{l} \underset{\gamma}{\text{minimize}} & \|L_{\gamma}\|_{2}^{2} \\ \text{subject to} & W_{\gamma} = z, \end{array} \tag{10}$$

where

$$\gamma = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \ L = \begin{bmatrix} Q^{1/2} Y K \\ R^{1/2} U K \end{bmatrix}, \ W = \begin{bmatrix} X_0 \\ Y_F \end{bmatrix} K, \ z = \begin{bmatrix} x_0 \\ y_f \end{bmatrix}$$

which admits the solution

 $\gamma^{\star} = (I - K_W (LK_W)^{\dagger} L) W^{\dagger} z,$

with K_W = Basis(Ker(W)). This leads to the following data-driven solution to Problem (8).

Theorem 3.1 (Data-driven LQ Control). The input \mathbf{u}_T^* that solves the LQ control problem (8) is

$$\mathbf{u}_{T}^{*} = UK \underbrace{(I - K_{W}(LK_{W})^{\dagger}L)W^{\dagger} \begin{bmatrix} x_{0} \\ y_{f} \end{bmatrix}}_{y_{f}^{*}}.$$
(11)

With the additional assumption that $\operatorname{Rank}(Q^{1/2}O_T^Y) = n,^3$ an alternative expression for \mathbf{u}_T^* that does not require the computation of kernel matrices is extracted from

$$\begin{bmatrix} \mathbf{x}_0\\ \mathbf{u}_T^* \end{bmatrix} = P^{-\frac{1}{2}} \left(\begin{bmatrix} X_0\\ Y_F \end{bmatrix} \begin{bmatrix} X_0\\ U \end{bmatrix}^{\dagger} P^{-\frac{1}{2}} \right)^{\dagger} \begin{bmatrix} x_0\\ y_f \end{bmatrix},$$
(12)

where

$$P = \left(Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger}\right)^{\dagger} Q \left(Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger}\right) + \begin{bmatrix} 0 & 0 \\ 0 & R \end{bmatrix}.$$
 (13)

A proof of (12) is postponed to the Appendix. We remark that (11) and (12) rely on assumption (4). In practice, this imposes a lower bound on the number N of trajectories recorded in (2), i.e., $N \ge mT + n$. In fact, when this condition is not satisfied, then the Problem (10) may be infeasible or not return the desired optimal control for some choices of the initial condition x_0 and target output y_f .

Remark 3 (Closed-form Solutions vs. Iterative and Optimization-Based Solutions). The distinctive aspect of the above results lies in the utilization of a closed-form formula, as shown in Theorem 3.1 and in most of the results discussed in this review. While exceptions do exist, such as those in Pellegrino, Blanchini, Fenu, and Salvato (2023a, 2023b) and da Silva, Bazanella, Lorenzini, and Campestrini (2018), it is noteworthy that much of the existing literature on data-driven control predominantly revolves around iterative and optimization-based methodologies. It is essential to acknowledge that certain problems necessitate optimization-based approaches, often those involving nonlinear systems or input-output constraints. Nevertheless, when the opportunity for closed-form solutions arises, closed-form expressions provide not only valuable insights into problem solutions but also confer numerical advantages over alternative methods, see e.g., Celi et al. (2022) for a discussion on the computational advantages of closed-form formulas over optimization-based solutions. Further, closed-form formulas have been recently used in the context of data-driven control to solve a set of diverse problems, such as the computation of the Kalman Filter and of the Linear Quadratic Gaussian regulator (Al Makdah & Pasqualetti, 2023), the identification of geometric invariant subspaces (Celi & Pasqualetti, 2022), and the solution of the eigenstructure assignment problem (Celi, Baggio, & Pasqualetti, 2023a).



Fig. 1. This figure shows the output trajectory of the system in Example 1 with \mathbf{u}_T computed through (11). Data is collected as described in (2), with X_0 and U random matrices with i.i.d. entries. The experiment is run with Q_t = blk-diag(10, 10, 10, 100, 100, 100), $R_t = I_m$, for all $t = \{1, ..., T\}$, and T = 150. Panel (a) shows the trajectories of the position (solid lines) and the orientation angles (dashed lines). Panel (b) shows the trajectory of the position in 3-D space (circle: initial position, cross: final position).

Example 1. We now apply the result from Theorem 3.1 to control a simplified model of a quadcopter. The quadcopter is dimensioned after a Crazyflie drone, with data (2) collected using Matlab simulations. For more details on the model of the system, we refer the reader to Wang, Man, Cao, Zheng, and Zhao (2016). The state of the system is

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} & \theta & \dot{\theta} & y & \dot{y} & \phi & \dot{\phi} & z & \dot{z} & \psi & \dot{\psi} \end{bmatrix}^{\mathsf{T}},$$

where (x, y, z) and $(\dot{x}, \dot{y}, \dot{z})$ are the coordinates and linear velocities, respectively, and (θ, ϕ, ψ) and $(\dot{\theta}, \dot{\phi}, \dot{\psi})$ are the roll, pitch and yaw angles and their respective velocities. The measured outputs are limited to the coordinates (x, y, z) and the asset angles (θ, ϕ, ψ) . The system is simulated in Fig. 1, where \mathbf{u}_T is obtained from (11).

3.1. Minimum energy control

T 1

The minimum-energy control problem (ME) is obtained from (8) by letting $Q_t = 0$ and $R_t = I$. It reads as (Kailath, 1980; Pasqualetti, Zampieri, & Bullo, 2014)

$$\min_{u} \sum_{t=0}^{N-1} \|u(t)\|_{2}^{2}$$
s.t. $x(t+1) = Ax(t) + Bu(t),$ (14)
 $y(t) = Cx(t) + Du(t),$
 $x(0) = x_{0}, \quad y(T-1) = y_{f}.$

Clearly, a solution to this problem can be obtained from Theorem 3.1 by simply letting L = UK. An insightful instance of the minimum energy control problem (14) is when $x_0 = 0$ and $X_0 = 0$, that is, the initial conditions of the experimental data and of the control problem are all equal to zero. In this case, although Lemma 2.1 cannot be used for a data-based representation of the system trajectories since X_0 is not fullrow rank, a direct expression for the minimum energy input can readily be obtained.

Theorem 3.2 (Data-driven ME Control Baggio et al., 2019). Let $X_0 = 0$ and Rank(U) = mT. Then, the input \mathbf{u}_T^* that solves the minimum energy control problem (14) is

$$\mathbf{u}_{T}^{*} = U(I - K_{Y_{\rm F}}(UK_{Y_{\rm F}})^{\dagger}U)Y_{\rm F}^{\dagger}y_{\rm f}, \qquad (15)$$

where Y_F are the final states of the recorded trajectories (3) and K_{Y_F} = Basis(Ker(Y_F)).

Eq. (15) is obtained from (11) leveraging the simplifications due to $X_0 = 0$. Along the lines of the derivation of (12), the following alternative minimum-energy control expression holds:

$$\mathbf{u}_T^* = (Y_F U^\dagger)^\dagger y_f. \tag{16}$$

 $^{^3}$ This is a mild condition that is satisfied, for instance, when (1) is observable and Q > 0.



Fig. 2. Panels (a)–(b) show the norm of the minimum energy control computed through: (i) the model-based formula $\mathbf{u}_T = C_T^{\dagger}(x_t - A^T x_0)$, where C_T is the controllability matrix defined in Lemma 2.2, (ii) the controllability Gramian with $u(t) = B^{T}(A^{T})^{T-t-1}W_T^{\dagger}(x_t - A^T x_0)$, (iii) the exact data-driven expression (16), and (iv) the asymptotically correct expression (19). The underlying system is generated randomly, with n = 20, m = 2, T = 40, $x_t = [11 \dots 1]^{T}$ and N is specified on the *x*-axis. The curves represent the average over 100 experiments with input data generated as random i.i.d. normal entries. Notice that, in accordance with the conditions imposed by Lemma 2.1, the expression (15) becomes exact once N = n + mT = 80 linearly independent experimental trajectories have been collected. Further, notice that (19) returns a feasible (the control satisfies the constraint) yet suboptimal (the cost achieved by the control is not optimal) solution for finite values of N. Panels (c)–(d) show the norm of the inputs \mathbf{u}_T computed as above, and the corresponding errors in the final state, as a function of the system dimension n. We note that while the Gramian-based solution is technically exact, it is effectively less robust than the exact data-driven formulas in (16) and (19) for systems with a large number of states. This loss of numerical precision is a consequence of the computational process involved in computing and manipulating the controllability Gramian. We let m = 2, T = n, N = mT + 20, and $x_t = [11 \dots 1]^T$. The matrices A and B are generated randomly. The curves represent the average over 100 experiments with data generated as above.

When C = I and D = 0 this formula offers a data-driven way to compute the *T*-step Gramian and its eigenvalues. In fact, it can be seen that in this case Eq. (16) can be written as a function of $X_{\rm F}$ and⁴ the controllability matrix in (7) equals $C_T = X_{\rm F}U^{\dagger}$. Consequently, the Gramian satisfies

$$W_T = C_T C_T^{\mathsf{T}} = X_{\mathsf{F}} U^{\dagger} U^{\dagger}^{\mathsf{T}} X_{\mathsf{F}}^{\mathsf{T}}.$$
(17)

Similarly, since the smallest (resp. largest) eigenvalues of the Gramian identify the states that require largest (resp. smallest) input energy, these can also be computed as

$$\sigma_{\min}^{-1}(W_T) = \max_{\|x_F\|_2 = 1} \|(X_F U^{\dagger})^{\dagger} x_F\|_2^2 = \sigma_{\max}^2((X_F U^{\dagger})^{\dagger}),$$

$$\sigma_{\max}^{-1}(W_T) = \min_{\|x_F\|_2 = 1} \|(X_F U^{\dagger})^{\dagger} x_F\|_2^2 = \sigma_{\min}^2((X_F U^{\dagger})^{\dagger}).$$
(18)

Finally, when the entries of U are i.i.d. random variables with zero mean and nonzero finite variance, one can obtained simplified expressions of the minimum energy control input. For instance, the simplified expression

$$\hat{\mathbf{u}}_T = U X_F^{\dagger} x_f \tag{19}$$

converges to the minimum-energy input \mathbf{u}_{T}^{*} almost surely, as the number of trajectories N increases (Baggio et al., 2019). Since the minimum energy input is unique and for any finite value of N it generally holds $\hat{\mathbf{u}}_T \neq \mathbf{u}_T^*$, it follows that (19) is a suboptimal input for the minimum energy problem (14). In particular, the input (19) drives the system to the desired final state $x_{\rm f}$ with non-minimum energy, while requiring fewer numerical operations for its computation when compared with (15), (16). This property can be useful when dealing with large (network) systems, for which these calculations are generally ill-conditioned (Baggio et al., 2021, 2019; Pasqualetti et al., 2014). In support of these claims, in Fig. 2 we perform a series of numerical experiments, where we assess the numerical robustness and accuracy of direct data-driven controls. We notice that the accuracy in computing the minimum-energy control input using the data-driven expression (16) is comparable to that achieved when using the modelbased counterpart, yet numerically more accurate than the model-based Gramian formula. Further, in Fig. 2(c)–(d) we notice that the accuracy of the Gramian-based control input decreases as n increases, while the data-driven expressions of the minimum-energy control inputs remain accurate for systems of considerably larger dimension.

As we progress through this paper, the closed-form expressions outlined in this section will form the foundational building blocks for deriving novel closed-form solutions to a variety of problems. Among these, in the following sections we will cover the infinitehorizon linear quadratic regulator problem (Celi et al., 2022), the linear quadratic Gaussian control problem (Al Makdah & Pasqualetti, 2023), the distributed data-driven linear quadratic control problem (Celi et al., 2023b). Further, we will answer several robustness questions related to data-driven control (Anguluri, Al Makdah, Katewa, & Pasqualetti, 2020; Celi et al., 2023b).

3.2. Datasets with heterogeneous control horizons

The results presented thus far assume that the experimental trajectories have the same length. In practice, however, it may be more convenient to collect data from heterogeneous experiments with different control horizons. The first question that we answer is whether the trajectories of the system can be represented as a linear combination of trajectories of different lengths, and for which control horizon. Is it possible to represent trajectories of length greater than those of the control experiments? Ultimately, this analysis leads to an extension of Lemma 2.1.

To formalize the discussion, assume that the control experiments are performed using *M* distinct horizons $T_i \in \mathbb{N}, i \in \{1, ..., M\}$, and that the available data is organized as $(U_i, X_{0,i}, X_i, Y_i)$, $i \in \{1, ..., M\}$, where the *i*th set contains N_i experiments, and $X_{0,i} \in \mathbb{R}^{n \times N_i}$, $U_i \in \mathbb{R}^{mT_i \times N_i}$, $X_i \in \mathbb{R}^{nT_i \times N_i}$, and $Y_i \in \mathbb{R}^{pT_i \times N_i}$ denote the matrices containing the initial states of the experiments, and the input, state, and output sequences with horizon T_i . Finally, let $X_{\mathrm{F},i}$ contain the last *n* rows of X_i and $D_H = \{(U_i, X_{0,i}, X_i, Y_i)\}_{i=1}^M$ be the set of heterogeneous data. See Fig. 3 for an illustration of the heterogeneous dataset.

We now present an extension of Lemma 2.1 that allows us to represent trajectories of (1) of length *T* using the heterogeneous data D_H , where the horizon *T* is an integer-weighted combination of the experimental horizons T_i (e.g., if $T_1 = 2$, $T_2 = 3$, and $T_3 = 8$, we could take $T = T_2 + 2T_3 = 19$). The main ideas behind this result are that a trajectory of length *T* can be broken up in multiple sub-trajectories, where the initial state of each sub-trajectory equals the final state of the previous sub-trajectory, and each sub-trajectory admits a data-based representation as in Lemma 2.1 (see also Fig. 4). We remark that the decomposition of a trajectory into multiple parts of a certain length may not be unique, thus leading to potentially multiple data-driven representations of the same trajectory when using heterogeneous data (for instance, if $T_1 = 2$, $T_2 = 3$, and $T_3 = 8$, then we could take $T = 19 = 8T_1 + T_3 = T_2 + 2T_3 = T_3 + T_2 + T_3$).

⁴ In this case the terminal constraint for (14) becomes $x(T) = x_f$.



Fig. 3. This figure shows the data collection phase with heterogeneous datasets.



Fig. 4. This figure shows an example of decomposition of a input-state trajectory in sub-trajectories: a trajectory of length T = 10 is divided in three sub-trajectories of lengths $T_1 = 4$, $T_2 = 2$, $T_3 = 4$.

Lemma 3.3 (Data-based Free and Forced Response Representation with Heterogeneous Data). Let D_H be the set of heterogeneous data and assume that

$$\operatorname{Rank}\left(\begin{bmatrix} X_{0,i} \\ U_i \end{bmatrix}\right) = mT_i + n.$$
(20)

Let ℓ_1, \ldots, ℓ_p be a sequence of indices such that $T = \sum_{i=1}^p T_{\ell_i}$. Further, let $K_{0,i} = \text{Basis}(\text{Ker}(X_{0,i})), K_{U,i} = \text{Basis}(\text{Ker}(U_i)),$

$$V_{i} = (X_{0,\ell_{i+1}} K_{U,\ell_{i+1}})^{\dagger} X_{F,\ell_{i}} K_{U,\ell_{i}}$$
$$Z_{i} = (X_{0,\ell_{i+1}} K_{U,\ell_{i+1}})^{\dagger} X_{F,\ell_{i}} K_{0,\ell_{i}}.$$

Then, for any initial state x_0 and input u_T , there exist vectors α and β such that

$$x_0 = X_{0,\ell_1} K_{U,\ell_1} \alpha, \text{ and} u_T = \text{blk-diag}(U_{\ell_1} K_{0,\ell_1}, \dots, U_{\ell_n} K_{0,\ell_n})\beta.$$

Moreover, \mathbf{x}_T and \mathbf{y}_T in (21) (see Box I) are the state and output trajectories of length T of (1) generated by x_0 and u_T .

A proof of Lemma 3.3 can be found in Appendix. Intuitively, the data-based representation of the *T*-steps state and output trajectories

of Lemma 3.3 is obtained by suitably "gluing" together the databased representations of system trajectories of lengths $\{T_1, \ldots, T_M\}$. Note in particular that the horizon *T* can be longer than the horizons of the experimental trajectories, thus allowing for the representation of trajectories that have never been observed during the experiments. As a special case, when the experimental trajectories have the same horizon, that is M = 1, it is possible to reconstruct from data system trajectories with horizons equal to any multiple integer of T_1 . This is illustrated in the next example.

Example 2 (Data-based Representation of Trajectories Longer Than the Experimental Data as in Lemma 3.3). Consider the scalar system

$$x(t+1) = ax(t) + u(t), \quad a \in \mathbb{R}$$

and the dataset with horizon T = 2

$$U_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, X_{0,1} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, X_1 = \begin{bmatrix} a & 1 & 0 \\ a^2 & a & 1 \end{bmatrix}.$$

It holds

1

$$K_{U,1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \ K_{0,1} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \ V_1 = a^2, \ Z_1 = \begin{bmatrix} a & 1 \end{bmatrix}$$

From Lemma 3.3, for any x_0 and \mathbf{u}_4 , there exist α and β such that

$$x_0 = X_{0,1}K_{U,1}\alpha = \alpha$$
, and
 $\mathbf{u}_4 = \text{blk-diag}(U_1K_{0,1}, U_1K_{0,1})\beta = \beta$,

and the state trajectory with initial condition x_0 and input sequence \mathbf{u}_4 in the interval [1,4] is given by

$$\mathbf{x}_{4} = \begin{bmatrix} X_{1}K_{U,1} & X_{1}K_{0,1} & 0\\ X_{1}K_{U,1}V_{1} & X_{1}K_{U,1}Z_{1} & X_{1}K_{0,1} \end{bmatrix} \begin{bmatrix} \alpha\\ \beta \end{bmatrix}$$
$$= \begin{bmatrix} a & 1 & 0 & 0 & 0\\ \frac{a^{2}}{a^{3}} & a^{2} & a & 1 & 0\\ \frac{a^{4}}{a^{3}} & a^{2} & a & 1 & 0\\ a^{4} & a^{3} & a^{2} & a & 1 \end{bmatrix} \begin{bmatrix} \alpha\\ \beta \end{bmatrix}. \Box$$

In addition to providing a new data-based representation of the trajectories of (1) that allows for multiple heterogeneous experiments, Lemma 3.3 can also be used to derive data-driven formulas that solve LQ control problems, among others. For instance, under the assumption of Lemma 3.3, the data-driven LQ control expression in (3.1) remains valid when the dataset D_H is used by redefining *L* and *W* as follows:

$$L = \begin{bmatrix} Q^{1/2} Y_{\ell_{1:p}} \\ R^{1/2} U_{\ell_{1:p}} \end{bmatrix}, \ W = \begin{bmatrix} X_{0,\ell_1} K_{U,\ell_1} \\ Y_{\ell_{1:p},F} \end{bmatrix},$$

where $U_{\ell_{1:p}} = \text{blk-diag}(U_{\ell_1}K_{0,\ell_1}, \dots, U_{\ell_M}K_{0,\ell_p})$, and $Y_{\ell_{1:p}}$ and $Y_{\ell_{1:p},F}$ denote the matrices consisting of the last *pT* and *p* rows, respectively, of the matrix in (21).

Remark 4 (*Alternative Approaches to Using Heterogeneous Datasets*). Alternative data-based representations of system trajectories with heterogeneous datasets have been proposed in Baggio and Pasqualetti (2020) and van Waarde, De Persis et al. (2020). In Baggio and Pasqualetti (2020), the experimental data consist of inputs, initial and final state recordings with different time horizons T_i . The resulting data-based representation of state trajectories is given in sampled form and exploits a data-based reconstruction of A^{T_i} . In van Waarde, De Persis et al. (2020) a generalization of Willems' Fundamental Lemma to the case of multiple trajectories is presented. Differently from our approach, this generalization does not seem to allow for the computation of trajectories longer than the experiments. \Box

$$\begin{bmatrix} \mathbf{x}_{\ell_{1}} K_{U,\ell_{1}} & X_{\ell_{1}} K_{0,\ell_{1}} & 0 & \cdots & \cdots & 0 \\ X_{\ell_{2}} K_{U,\ell_{2}} V_{1} & X_{\ell_{2}} K_{U,\ell_{2}} Z_{1} & X_{\ell_{2}} K_{0,\ell_{2}} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{\ell_{p}} K_{U,\ell_{p}} V_{p-1} \cdots V_{1} & X_{\ell_{p}} K_{U,\ell_{p}} Z_{p-1} \cdots Z_{1} & X_{\ell_{p}} K_{U,\ell_{p}} Z_{M-2} \cdots Z_{1} & \cdots & X_{\ell_{p}} K_{0,\ell_{p}} Z_{1} & X_{\ell_{p}} K_{0,\ell_{p}} \\ Y_{\ell_{1}} K_{U,\ell_{1}} & Y_{\ell_{1}} K_{0,\ell_{1}} & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{\ell_{p}} K_{U,\ell_{p}} V_{M-1} \cdots V_{1} & Y_{\ell_{p}} K_{U,\ell_{p}} Z_{M-1} \cdots Z_{1} & Y_{\ell_{p}} K_{U,\ell_{p}} Z_{M-2} \cdots Z_{1} & \cdots & Y_{\ell_{p}} K_{0,\ell_{p}} Z_{1} & Y_{\ell_{p}} K_{0,\ell_{p}} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

$$(21)$$

Box I.

4. Robustness of data-driven open-loop LQ control

One of the main advantages of the direct data-driven approach with explicit formulas presented thus far is the possibility to analytically explore the impact of perturbations on these formulas and easily compute their sensitivity to parameter variations. In fact, such an analysis yields a way to quantify and improve the robustness of the direct datadriven approach to noisy and corrupted datasets. We consider two cases, which differ in the available knowledge of the statistics of the perturbations affecting the data. Let the perturbed dataset be

$$\tilde{U} = U + \Delta_U, \tag{22a}$$

$$\tilde{X}_0 = X_0 + \Delta_0, \tag{22b}$$

$$\tilde{X} = X + \Delta_X, \tag{22c}$$

$$\tilde{Y} = Y + \Delta_Y, \tag{22d}$$

where U, X_0 , X, Y denote the ground truth values as in (2) and Δ_U , Δ_0 , Δ_X , Δ_Y contain stochastic perturbations. For the purpose of this discussion, we focus on the data-driven control in (12). However, the analysis can be adapted to other data-driven control expressions.

4.1. Data-driven LQ control with known noise statistics

We start with the standard scenario of i.i.d. perturbations characterized by known second-order statistics. In particular, we assume that Δ_U , Δ_0 , Δ_Y are random matrices consisting of i.i.d. entries with zero mean and variance σ_U^2 , σ_0^2 , and σ_Y^2 , respectively. In this setting, the data-driven control (12) is not consistent; that is, it does not converge to the true optimal control input even when the amount of available data grows to infinity. This lack of consistency is due to the pseudoinverse operation

$$\begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\dagger} = \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \right)^{\dagger},$$

which contains quadratic terms (namely, $\tilde{X}_0 \tilde{X}_0^{\mathsf{T}}$, $\tilde{U}\tilde{U}^{\mathsf{T}}$) that, by the law of large numbers, introduce variance-dependent biases (namely, $\sigma_0^2 NI$ and $\sigma_U^2 NI$) as *N* grows. However, since the noise variances are known, we can include correction terms in (12) that compensate for these biases and achieve asymptotically accurate data-driven formulas. Specifically, (12) can be modified as follows:

$$\begin{bmatrix} x_0^c \\ \mathbf{u}_c^r \end{bmatrix} = (P_c^{\dagger})^{\frac{1}{2}} \left(\begin{bmatrix} I_n & 0 \\ \tilde{Y}_F \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}_c^{\dagger} \right) (P_c^{\dagger})^{\frac{1}{2}} \right)^{\prime} \begin{bmatrix} x_0 \\ y_f \end{bmatrix},$$
(23) where

. +

$$P_{c} = \left(\tilde{Y} \begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix}_{c}^{\dagger} \right)^{\mathsf{T}} \mathcal{Q} \left(\tilde{Y} \begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix}_{c}^{\dagger} \right) + \begin{bmatrix} 0 & 0 \\ 0 & R \end{bmatrix}, \\ \begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix}_{c}^{\dagger} = \begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix} \begin{bmatrix} \tilde{X}_{0} \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} - \begin{bmatrix} \sigma_{0}^{2} N I_{n} & 0 \\ 0 & \sigma_{U}^{2} N I_{mT} \end{bmatrix} \right)^{\dagger}$$

_



Fig. 5. This figure shows the asymptotic consistency of 23 for problem (8). The left panel shows how the controller computed with a noisy dataset with noise compensation in 23 (solid lines) asymptotically converges to the optimal controller \mathbf{u}_T^* for (8), as opposed to the non-compensated expression in (12) (dashed lines), for varying noise statistics. The experiment is performed over a randomly generated system with n = 5, m = 3, p = 2 and T = 15. Finally, $\sigma_0 = \sigma_Y = 0.1$, while different values of σ_U are shown in the legend. The right panel further supports this result by showing how the difference between *P* and P_c evolves as the number of experiment increases and the noise in the data is appropriately accounted for as shown in Theorem 4.1 (here, $\sigma_{tt} = 0.5$).

and the following consistency result holds (see the Appendix for the proof).

Theorem 4.1 (Asymptotic Consistency of 23). Assume that

- (i) the columns of X₀ and U are i.i.d. and satisfy the condition (4) almost surely as N → ∞, and
- (ii) the entries of Δ_U , Δ_0 , and Δ_Y are i.i.d. with zero mean and variances σ_U^2 , σ_0^2 , σ_Y^2 .

Then, \mathbf{u}_T^c in 23 converges almost surely to \mathbf{u}_T^* as $N \to \infty$.

Fig. 5 shows the asymptotic consistency of 23, as predicted by Theorem 4.1, for a randomly generated system.

4.2. Data-driven LQ control with unknown noise statistics

In the more general case of (possibly correlated) perturbations with unknown second-order statistics, the robustness of (12) can be assessed through a local sensitivity analysis. Here, we assume for simplicity that the noise acts on the output measurements only, that is $\Delta_0 = \Delta_U = 0$. However, an analysis similar to the one that follows can be carried out also for noisy inputs and states. Let

$$F(U, X_0, Y) = (P^{\dagger})^{\frac{1}{2}} \left(\begin{bmatrix} X_0 \\ Y_F \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger} (P^{\dagger})^{\frac{1}{2}} \right)^{\dagger} \begin{bmatrix} x_0 \\ y_f \end{bmatrix}$$
(24)

denote the data-driven control map (12) with *P* as in (13), and $\operatorname{supp}(\Delta_Y) = \{i : \delta_{Y,i} \neq 0\}$, with $\delta_{Y,i} = \operatorname{vec}(\Delta_Y)_i$, denote the set of corrupted entries of *Y*. Since *F* is Fréchet-differentiable with respect to

Y at the ground truth data, we can write it through its Taylor expansion as

$$F(U, X_0, \tilde{Y}) = F(U, X_0, Y) + \nabla F_Y(U, X_0, Y) \operatorname{vec}(\Delta_Y) + r(U, X_0, Y, \Delta_Y),$$
(25)

with $\lim_{\|\Delta_Y\|_2 \to 0} \|r(U, X_0, Y, \Delta_Y)\|_2 / \|\Delta_Y\|_2 = 0$ and where $\nabla F_Y(U, X_0, Y)$ is the Jacobian matrix of *F* with respect to *Y* calculated at the ground truth data. If the expected norms of the perturbations are sufficiently small, then (25) can be well approximated as (see the Appendix)

$$F(U, X_0, \tilde{Y}) \approx F(U, X_0, Y) + \nabla F_Y(U, X_0, Y) \operatorname{vec}(\Delta_Y).$$
(26)

For notational convenience, let $\nabla F_{Y,i}$ be the *i*th column of ∇F_Y (U, X_0, Y), and let

$$\Delta y_{\rm f} = \|\tilde{y}_{\rm f} - y_{\rm f}\|_2$$

measure the error induced by the noisy data on the final output \tilde{y}_f when using the data-driven control input in (26). We next investigate how the sensitivity of the data-driven map, as quantified by the norm of the Jacobian matrix ∇F_Y , is related to the data size *N*.

Lemma 4.2 (Properties of $\|\nabla F_Y\|_2$ as a Function of N). Assume that the entries of X_0 , U are independent⁵ of N and that $\sigma^2_{\min}[(X_0^{\mathsf{T}} U^{\mathsf{T}}]^{\mathsf{T}}) \ge cN$ where c > 0 is a constant independent of N. Then, for all $i \in \operatorname{supp}(\Delta_Y)$, $\|\nabla F_{Y,i}\|_2 \le k_{Y,i}/N$, where $k_{Y,i} > 0$ are constants independent of N.

The condition $\sigma_{\min}^2([X_0^{\mathsf{T}} U^{\mathsf{T}}]^{\mathsf{T}}) \ge cN$ is typically satisfied for random i.i.d. initial conditions and inputs.⁶ Thus, Lemma 4.2 shows that all $\|\nabla F_{Y,i}\|_2$ typically converge to zero as the number of experiments *N* increases. Under this scenario, as additional data become available, the map *F* becomes increasingly more robust against corrupted data. This conclusion is instrumental for the following result, whose proof is postponed to Appendix.

Theorem 4.3 (Asymptotic Robustness for Sublinear Number of Perturbations). In addition to the assumptions in Lemma 4.2, assume also that the entries of Δ_Y are independent of N. Then, if the cardinality of supp (Δ_Y) grows sublinearly⁷ with N, for any $\tau > 0$,

$$\lim_{N \to \infty} \mathbb{P}\left[\Delta y_{\mathrm{f}} \ge \tau\right] = 0. \tag{27}$$

Theorem 4.3 guarantees that the error in the final output decreases to zero when *N* increases, regardless of Δ_Y (see Fig. 6 for a numerical example). This ensures the robustness of the data-driven control action for small, possibly adversarial, perturbations.

Remark 5 (*Comparison with Existing Approaches*). Numerous studies have focused on developing data-driven controllers that can effectively handle disturbances generated by worst-case or stochastic noise models. Most of the existing approaches rely on robustified versions of data-based optimization problems, typically achieved through suitable regularizations (e.g., see Berberich, Köhler, Müller, & Allgöwer, 2020; Breschi, Chiuso, & Formentin, 2023; Coulson, Lygeros, & Dörfler, 2021; De Persis & Tesi, 2021; Dörfler, Tesi, & De Persis, 2023) or by leveraging classic robust control tools (e.g., see Berberich, Scherer, & Allgöwer, 2022; Bisoffi, De Persis, & Tesi, 2021, 2022; van Waarde, Camlibel and Mesbahi, 2020). The distinctive feature of the approach of this paper lies in the use of closed-form expressions which allows to asymptotically compensate the influence of noise and to explicitly characterize the sensitivity of data-driven controls. □



Fig. 6. This figure shows the convergence results of Theorem 4.3. As implied by the theorem, although the noise statistics remain unknown, the difference between the desired final output and that computed with a noisy dataset approaches zero as the number of experiments increases. The experiment is performed over a randomly generated system with n = 5, m = 3, p = 2 and T = 15. The noise on the output is additive, gaussian, and with distribution $\mathcal{N}(1, 0.5)$.

5. Data-driven formulas for closed-loop LQ control

In this section we study the LQ control problem

$$\begin{array}{ll} \underset{u,x}{\text{minimize}} & \sum_{t=0}^{T-1} \left(\|x(t)\|_{Q_t}^2 + \|u(t)\|_{R_t}^2 \right) \\ \text{subject to} & x(t+1) = Ax(t) + Bu(t), \\ & x(0) = x_0. \end{array}$$
(28)

Differently from Problem (8), Problem (28) does not impose a terminal constraint on the output trajectory and it thus allows for a solution in feedback form. In particular, the (model-based) input that solves Problem (28) is

$$u(t) = \underbrace{-(R_t + B^{\mathsf{T}} P_{t+1} B)^{-1} B^{\mathsf{T}} P_{t+1} A}_{K_{LQR}^t} x(t),$$
(29)

where P_t satisfies the Riccati equation

$$P_{t-1} = Q_t + A^{\mathsf{T}} P_t A - A^{\mathsf{T}} P_t B (R_t + B^{\mathsf{T}} P_t B)^{-1} B^{\mathsf{T}} P_t A,$$
(30)

and $P_T = 0$ (Lancaster & Rodman, 1995). As a known result, when Q and R are time-invariant and $T = \infty$, the time-varying control law (29) reduces to a time-invariant control law, in which case the solution to Problem (28) becomes

$$u(t) = K_{\rm LQR} x(t)$$

In this section we show how a dataset of finite length can be used to compute the time-varying feedback K_{LQR}^t as well as to estimate the static feedback K_{LQR} . The main ideas behind our approach are that (i) optimal open-loop input and state trajectories that solve the LQ problem (28) can be computed from data, similarly to the approach in Section 3, and (ii) the feedback gains K_{LQR}^t can be computed solving linear regression problems between optimal open-loop input and state trajectories.

Theorem 5.1 (Data-driven Finite-Horizon LQR Gains). Let U_t and X_t be the submatrices of U and X in (2) obtained by selecting only the inputs and states at time t, and define the matrices $L = \left[(Q^{1/2}XK)^{\mathsf{T}} \quad (R^{1/2}UK)^{\mathsf{T}} \right]^{\mathsf{T}}$, and

$$\mathbf{U}_t = U_t K (I - K_W (LK_W)^{\dagger} L) W^{\dagger}, \qquad (31)$$

$$\mathbf{X}_{t} = \begin{cases} I, & t = 0, \\ X_{t}K(I - K_{W}(LK_{W})^{\dagger}L)W^{\dagger}, & t > 0, \end{cases}$$
(32)

⁵ We say that a random variable x is independent of a deterministic parameter α if the distribution of x is not a function of α .

⁶ If *U* and *X*₀ have i.i.d. entries with zero mean and variance σ^2 , $\frac{1}{N}\sigma_{\min}^2([X_0^\top U^\top]^\top)$ tends almost surely to σ^2 as *N* tends to infinity by the law of large numbers.

⁷ A sequence $\{x_n\}$ grows sublinearly with *n* if $\lim_{n\to\infty} x_n/n = 0$.



Fig. 7. This figure supports the results of Section 5. In particular, we compute K_{LQR}^0 using Theorem 5.1 for increasing *T*. As expected from (34) the distance between U_0 and K_{LQR} , computed as the 2-norm of their difference, decreases as *T* grows. The experiment is performed over a randomly generated system with n = 5 and m = 2.

with
$$W = X_0 K$$
 and $K_W = \text{Basis}(\text{Ker}(W))$. Then,⁸
 $K_{\text{LOR}}^t = \mathbf{U}_t(\mathbf{X}_t)^{-1}$. (33)

A proof of Theorem 5.1 is postponed to Appendix. The above result allows us to compute any element of the sequence of time-varying controllers K_{LQR}^t as long as $t \le T$, where *T* is the horizon of the available dataset (2). Interestingly, Theorem 5.1 only uses forward trajectories of the system and provides a closed-form, explicit expression of the LQR gains, thus avoiding the use of recursive, implicit Riccati equations or backward-in-time dynamic programming. We highlight that K_{LQR}^0 converges to the steady state gain K_{LQR} as *T* increases (see also Fig. 7). In particular, we have

$$\|\mathbf{U}_{0} - K_{\text{LOR}}\|_{2} \le c\rho^{T},\tag{34}$$

where c > 0 and $0 < \rho < 1$ are suitable constants independent of *t* (Lancaster & Rodman, 1995).

Remark 6 (Related Work on Data-Driven LQ Control). Linear quadratic control has received the most attention in the recent data-driven control literature. As opposed to the direct and closed-formulas discussed in this paper and, e.g., in Pellegrino et al. (2023a, 2023b), most approaches in the literature rely on indirect schemes (Aangenent, Kostic, de Jager, van de Molengraft, & Steinbuch, 2005; da Silva et al., 2018), where a model of the system is first identified, optimization-based schemes (Coulson, Lygeros, & Dörfler, 2019; De Persis & Tesi, 2020; Dörfler, Tesi, & De Persis, 2022; Rotulo, De Persis, & Tesi, 2020), or iterative schemes inspired by reinforcement learning approaches (Abbasi-Yadkori, Lazic, & Szepesvári, 2019; Bradtke, Ydstie, & Barto, 1994; Dean, Mania, Matni, Recht, & Tu, 2020; Fazel, Ge, Kakade, & Mesbahi, 2018; Gravell, Esfahani, & Summers, 2020; Mohammadi, Zare, Soltanolkotabi, & Jovanović, 2019; Recht, 2019), among others. Alternative data-driven approaches to Theorem 5.1 to compute feedback gains can be found, among others, in Al Makdah et al. (2022), Celi and Pasqualetti (2022), De Persis and Tesi (2020) and Bianchin (2023), Celi et al. (2023a), which solve the general eigenstructure assignment problem via static feedback in a purely data-driven manner.

6. Conclusion and future work

This tutorial paper presents a framework to solve a variety of linear quadratic control problems for linear systems using a dataset of input, state, and output trajectories collected offline, and to analyze the robustness of these solutions to noise and arbitrary perturbations. Differently from approaches relying on system identification, optimization, and policy iteration, this paper presents a set of closed-form expressions for optimal and sub-optimal control sequences, which are computationally efficient, insightful, and enable a direct sensitivity analysis of data-driven control. In fact, in some cases, these formulas display favorable computational properties even when compared to classic model-based solutions, while also avoiding the solution of typically implicit and recursive Riccati equations. The paper contains also a number of numerical studies to validate the approach and showcase its tradeoffs.

This paper shows that the data-driven approach to control may offer solutions that are computationally advantageous with respect to classic methods in the state–space, frequency, or geometric approaches. However, a detailed analysis of when data-driven methods should be preferred to classic ones, a comprehensive comparison of direct and indirect methods, the extension beyond linear quadratic control, among others, remain outstanding timely questions and potentially interesting research avenues.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Appendix

A.1. Proof of the results in Section 2

We start with the following technical result.

Lemma A.1 (Rank of Block Matrices). Let $A \in \mathbb{R}^{n_1 \times m}$ and $B \in \mathbb{R}^{n_2 \times m}$, with $m \ge n_1 + n_2$. Let K_A = Basis(Ker(A)) and K_B = Basis(Ker(B)). The following statements are equivalent:

(i) Rank
$$\begin{pmatrix} A \\ B \end{pmatrix} = n_1 + n_2;$$

(ii) Rank $(AK_B) = n_1$ and Rank $(BK_A) = n_2$

Proof. ((ii) implies (i).) We will show that statement (ii) is violated when statement (i) is violated. Let vectors v_A and v_B satisfy $v_AA+v_BB = 0$, with $[v_A v_B] \neq 0$. Then, $v_AAK_B+v_BBK_B = v_AAK_B = 0$, which implies that either $v_A = 0$ or Rank $(AK_B) < n_1$. Similarly, $v_AAK_A + v_BBK_A = v_BBK_A = 0$, which implies that either $v_B = 0$ or Rank $(AK_B) < n_2$. Since v_A and v_B cannot be simultaneously zero, we conclude that statement (ii) implies (i).

((i) implies (ii).) Notice that

$$\operatorname{Rank}\left(\begin{bmatrix}A\\B\end{bmatrix}\right) = \operatorname{Rank}\left(\begin{bmatrix}A\\B\end{bmatrix}C\right),$$

with *C* any invertible matrix of appropriate dimension. Let $C = [K_B B^T]$. Then,

$$\begin{split} n_1 + n_2 &= \operatorname{Rank} \left(\begin{bmatrix} A \\ B \end{bmatrix} \right) = \operatorname{Rank} \left(\begin{bmatrix} A \\ B \end{bmatrix} \begin{bmatrix} K_B & B^{\mathsf{T}} \end{bmatrix} \right) \\ &= \operatorname{Rank} \left(\begin{bmatrix} A K_B & A B^{\mathsf{T}} \\ 0 & B B^{\mathsf{T}} \end{bmatrix} \right), \end{split}$$

which implies that $\operatorname{Rank}(AK_B) = n_1$. Repeating the reasoning with $C = [K_A \ A^T]$ concludes the proof.

We are now ready to prove Lemma 2.1.

⁸ We assume here that the matrix *A* is invertible, which guarantees that X_t is also invertible (see Appendix). When X_t is not invertible, the gain $U_t(X_t)^{\dagger}$ generates the input sequence that solves (28), but it may differ from the gain K_{LQR}^t computed as in (29).

Proof of Lemma 2.1. Using (4) and Lemma A.1, the matrices X_0K_U and UK_0 are full-row rank. Hence, for every x_0 and \mathbf{u}_T there exist α and β such that $x_0 = X_0K_U\alpha$ and $\mathbf{u}_T = UK_0\beta$. Notice that the data matrices satisfy the relations

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} O_T^X & F_T^X \\ O_T^Y & F_T^Y \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix},$$
(35)

where

$$\begin{split} O_T^X = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^T \end{bmatrix}, & F_T^X = \begin{bmatrix} B & \cdots & 0 & 0 \\ AB & \cdots & 0 & 0 \\ \vdots \\ A^{T-1}B & \cdots & AB & B \end{bmatrix}, \\ O_T^Y = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{T-1} \end{bmatrix}, & F_T^Y = \begin{bmatrix} D & \cdots & 0 & 0 \\ CB & \cdots & 0 & 0 \\ \vdots \\ CA^{T-2}B & \cdots & CB & D \end{bmatrix}. \end{split}$$

Notice that

$$\begin{split} XK_U &= O_T^X X_0 K_U + F_T^X U K_U = O_T^X X_0 K_U, \\ XK_0 &= O_T^X X_0 K_0 + F_T^X U K_0 = F_T^X U K_0. \end{split}$$

Then, the state trajectory x_T of (1) with input $\mathbf{u}_T = UK_0\beta$ and initial state $x_0 = X_0K_U\alpha$ can be written as

$$\mathbf{x}_T = O_T^X \mathbf{x}_0 + F_T^X \mathbf{u}_T = O_T^X X_0 K_U \alpha + F_T^X U K_0 \beta$$
$$= X K_U \alpha + X K_0 \beta.$$

The claimed expression for \mathbf{y}_T is obtained similarly using the matrices O_T^Y and F_T^Y , thus concluding the proof.

We now provide a proof of Lemma 2.2.

Proof of Lemma 2.2. Notice from (35) that $YK_U = O_T^Y X_0 K_U$. From Bernstein (2009, Fact 2.10.2), $Im(O_T^Y X_0 K_U) = Im(O_T^Y)$ since $X_0 K_U$ is full row-rank (cif. (4) and Lemma A.1). Hence, $Rank(O_T^Y) = Rank(YK_U)$. Similarly, $X_F K_0 = C_T U K_0$ and $U K_0$ is full row-rank, thus implying that $Rank(C_T) = Rank(X_F K_0)$, which concludes the proof.

A.2. Proof of the results in Section 3

Proof of Eq. (12). Problem (8) can be equivalently rewritten as

$$\begin{array}{c} \underset{\mathbf{u}_{T}}{\text{minimize}} & \left\| P^{\frac{1}{2}} \begin{bmatrix} x(0) \\ \mathbf{u}_{T} \end{bmatrix} \right\|_{2}^{2} \\ \text{subject to} \end{array}$$
(36)

$$\begin{bmatrix} I & 0\\ O_{\rm F}^{Y} & F_{\rm F}^{Y} \end{bmatrix} \begin{bmatrix} x(0)\\ \mathbf{u}_{T} \end{bmatrix} = \begin{bmatrix} x_{0}\\ y_{\rm f} \end{bmatrix}$$

where $O_{\rm F}^{Y}$ and $F_{\rm F}^{Y}$ denotes the last *p* rows of the matrices O_{T}^{Y} and F_{T}^{Y} , and

$$P = \begin{bmatrix} O_T^Y & F_T^Y \end{bmatrix}^{\mathsf{T}} Q \begin{bmatrix} O_T^Y & F_T^Y \end{bmatrix} + \begin{bmatrix} 0 & 0\\ 0 & R \end{bmatrix}$$
(37)

From the assumptions R > 0 and $\operatorname{Rank}(Q^{1/2}O_T^Y) = n$, it follows that P > 0 (see Celi et al. (2023b, Theorem III.1) for a proof of this fact). Then, by defining

$$v = P^{\frac{1}{2}} \begin{bmatrix} x(0) \\ \mathbf{u}_T \end{bmatrix},$$

we can rewrite (36) as

 $\begin{array}{ll} \underset{v}{\text{minimize}} & \|v\|_{2}^{2} \\ \text{subject to} \\ \begin{bmatrix} I & 0 \\ O_{\text{F}}^{Y} & F_{\text{F}}^{Y} \end{bmatrix} P^{-\frac{1}{2}}v = \begin{bmatrix} x_{0} \\ y_{\text{f}} \end{bmatrix}. \end{array}$ (38)

The minimizer of (38) is

$$v^* = \left(\begin{bmatrix} I & 0 \\ O_F^Y & F_F^Y \end{bmatrix} P^{-\frac{1}{2}} \right)^{\mathsf{T}} \begin{bmatrix} x_0 \\ y_f \end{bmatrix}$$

Thus, the solution \mathbf{u}_T^* to (36) satisfies

$$\begin{bmatrix} x_0 \\ \mathbf{u}_T^* \end{bmatrix} = P^{-\frac{1}{2}} v^* = P^{-\frac{1}{2}} \left(\begin{bmatrix} I & 0 \\ O_F^Y & F_F^Y \end{bmatrix} P^{-\frac{1}{2}} \right)^{\dagger} \begin{bmatrix} x_0 \\ y_f \end{bmatrix}$$

To conclude, note that, under assumption (4), P in (37) equals (13) and

$$\begin{bmatrix} I_n & 0\\ O_F^Y & F_F^Y \end{bmatrix} = \begin{bmatrix} X_0\\ Y_F \end{bmatrix} \begin{bmatrix} X_0\\ U \end{bmatrix}',$$

from which (12) follows.

Proof of Lemma 3.3. From Lemma 2.1, there exist vectors α and β_1 such that $x_0 = X_{0,\ell_1} K_{U,\ell_1} \alpha$ and $\mathbf{u}_{T_1} = U_{\ell_1} K_{0,\ell_1} \beta_1$, and the corresponding state trajectory in the interval $[1, T_{\ell_1}]$ is

$$\mathbf{x}_{T_{\ell_1}} = X_{\ell_1} K_{U,\ell_1} \alpha + X_{\ell_1} K_{0,\ell_1} \beta_1$$

In particular, it holds

$$x(T_{\ell_1}) = X_{F,\ell_1} K_{U,\ell_1} \alpha + X_{F,\ell_1} K_{0,\ell_1} \beta_1,$$

Next, since $X_{0,\ell_2}K_{U,\ell_2}$ has full row rank, we have

$$x(T_{\ell_1}) = X_{0,\ell_2} K_{U,\ell_2} \gamma$$

where

$$\begin{aligned} \gamma &= (X_{0,\ell_2} K_{U,\ell_2})^{\dagger} (X_{F,\ell_1} K_{U,\ell_1} \alpha + X_{F,\ell_1} K_{0,\ell_1} \beta_1) \\ &= V_1 \alpha + Z_1 \beta_1. \end{aligned}$$

Thus, using Lemma 2.1 again, there exist a vector β_2 such that the input in the interval $[0, T_{\ell_1} + T_{\ell_2} - 1]$ can be written as

$$\mathbf{u}_{T_{\ell_1}+T_{\ell_2}} = \begin{bmatrix} U_{\ell_1}K_{0,\ell_1} & 0\\ 0 & U_{\ell_2}K_{0,\ell_2} \end{bmatrix} \begin{bmatrix} \beta_1\\ \beta_2 \end{bmatrix},$$

and the corresponding state trajectory in $[1, T_{\ell_1} + T_{\ell_2}]$ as

$$\begin{split} \mathbf{x}_{T_{\ell_1}+T_{\ell_2}} &= \begin{bmatrix} X_{\ell_1} K_{U,\ell_1} \alpha + X_{\ell_1} K_{0,\ell_1} \beta_1 \\ X_{\ell_2} K_{U,\ell_2} \gamma + X_{\ell_2} K_{0,\ell_2} \beta_2 \end{bmatrix} \\ &= \begin{bmatrix} X_{\ell_1} K_{U,\ell_1} \alpha + X_{\ell_1} K_{0,\ell_1} \beta_1 \\ X_{\ell_2} K_{U,\ell_2} V_1 \alpha + X_{\ell_2} K_{U,\ell_2} Z_1 \beta_1 + X_{\ell_2} K_{0,\ell_2} \beta_2 \end{bmatrix} \\ &= \begin{bmatrix} X_{\ell_1} K_{U,\ell_1} & X_{\ell_1} K_{0,\ell_1} & 0 \\ X_{\ell_2} K_{U,\ell_2} V_1 & X_{\ell_2} K_{U,\ell_2} Z_1 & X_{\ell_2} K_{0,\ell_2} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \end{bmatrix}. \end{split}$$

The expression of \mathbf{x}_T in (21) follows by iterating the previous argument p times and collecting all β_i as $\beta = [\beta_1^{\mathsf{T}} \cdots \beta_p^{\mathsf{T}}]^{\mathsf{T}}$. A similar reasoning holds for the output trajectory \mathbf{y}_T .

A.3. Proof of the results in Section 4

Proof of Theorem 4.1. By the strong law of large numbers (Van der Vaart, 2000) and the assumptions (i)-(ii) on the experiments and noise, as *N* grows, the entries of $\frac{1}{N}X_0\Delta_0^{\mathsf{T}}$, $\frac{1}{N}X_0\Delta_U^{\mathsf{T}}$, $\frac{1}{N}U\Delta_0^{\mathsf{T}}$, $\frac{1}{N}U\Delta_U^{\mathsf{T}}$, $\frac{1}{N}\Delta_0\Delta_U^{\mathsf{T}}$ tend to zero almost surely, while $\frac{1}{N}\Delta_0\Delta_0^{\mathsf{T}}$ and $\frac{1}{N}\Delta_U\Delta_U^{\mathsf{T}}$ tend to $\sigma_0^2 I$, $\sigma_U^2 I$ almost surely. This implies that, as $N \to \infty$,

$$\frac{1}{N} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \xrightarrow{\text{a.s.}} \frac{1}{N} \begin{bmatrix} X_0 \\ U \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\mathsf{T}} + \begin{bmatrix} \sigma_0^2 I_n & 0 \\ 0 & \sigma_U^2 I_{mT} \end{bmatrix}, \tag{39}$$

Similarly, by the strong law of large numbers, the entries of $\frac{1}{N}Y \Delta_0^{\mathsf{T}}$, $\frac{1}{N}Y \Delta_U^{\mathsf{T}}$, $\frac{1}{N}U \Delta_Y^{\mathsf{T}}$, $\frac{1}{N}X_0 \Delta_Y^{\mathsf{T}}$, $\frac{1}{N}\Delta_U \Delta_Y^{\mathsf{T}}$, $\frac{1}{N}\Delta_0 \Delta_Y^{\mathsf{T}}$ tend to zero almost surely with N, so that, as $N \to \infty$,

$$\frac{1}{N}\tilde{Y}\begin{bmatrix}\tilde{X}_0\\\tilde{U}\end{bmatrix}^{\top} \xrightarrow{\text{a.s.}} \frac{1}{N}Y\begin{bmatrix}X_0\\U\end{bmatrix}^{\top}.$$
(40)

Eqs. (39) and (40) imply that, as $N \to \infty$,

$$\begin{split} \tilde{Y} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}_c^{\dagger} &= \frac{1}{N} \tilde{Y} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} \left(\frac{1}{N} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix} \begin{bmatrix} \tilde{X}_0 \\ \tilde{U} \end{bmatrix}^{\mathsf{T}} - \begin{bmatrix} \sigma_0^2 I_n & 0 \\ 0 & \sigma_U^2 I_{mT} \end{bmatrix} \right)^{\dagger} \\ \xrightarrow{\text{a.s.}} Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} X_0 \\ U \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\mathsf{T}} \right)^{\dagger} &= Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger}. \end{split}$$

× †

Since the map in (23) is a continuous function of \tilde{U} , \tilde{X}_0 , \tilde{Y} at $\Delta_U = \Delta_0 = \Delta_Y = 0$, from the previous equation and the continuous mapping theorem (Van der Vaart, 2000), as $N \to \infty$,

$$\begin{bmatrix} x_0^c \\ \mathbf{u}_T^c \end{bmatrix} \xrightarrow{\text{a.s.}} P^{-\frac{1}{2}} \left(\begin{bmatrix} I_n & 0 \\ Y_F \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger} \end{bmatrix} P^{-\frac{1}{2}} \right)^T \begin{bmatrix} x_0 \\ y_f \end{bmatrix}$$

$$= P^{-\frac{1}{2}} \left(\begin{bmatrix} X_0 \\ Y_F \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger} P^{-\frac{1}{2}} \right)^{\dagger} \begin{bmatrix} x_0 \\ y_f \end{bmatrix} = \begin{bmatrix} x_0 \\ \mathbf{u}_T^* \end{bmatrix},$$

_

where we used the identity $X_0 \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} I_n & 0 \end{bmatrix}$, which holds when $\begin{bmatrix} X_0 \end{bmatrix}$.

 $\begin{bmatrix} X_0 \\ U \end{bmatrix}$ has full row rank, and the fact that (4) is satisfied almost surely as $N \to \infty$.

The following result, whose proof follows from Anguluri et al. (2020, Lemma1), confirms that when the expected norm of the perturbation Δ_Y is small so is the residual *r* of the expansion (25).

Lemma A.2 (FirSt-Order Approximation of $F(U, X_0, \tilde{Y})$). Let $\nabla F_{Y,i}$ be the *i*th column of ∇F_Y and

$$\Sigma = F(U, X_0, \tilde{Y}) - F(U, X_0, Y) - \sum_{i \in \text{supp}(\Delta_Y)} \delta_{Y,i} \nabla F_{Y,i}(U, X_0, Y).$$

Then, for any $\tau > 0$,

$$\lim_{\mathbb{E}[\|\operatorname{vec}(\varDelta_Y)\|_2] \to 0} \mathbb{P}\left[\|\varSigma\|_2 \ge \tau \sqrt{\mathbb{E}[\|\operatorname{vec}(\varDelta_Y)\|_2]}\right] = 0.$$

Proof of Lemma 4.2. Let

$$G = \begin{bmatrix} X_0 \\ Y_F \end{bmatrix} \begin{bmatrix} X_0 \\ U \end{bmatrix}$$

and rewrite (24) as

$$\begin{split} F(U, X_0, Y) &= P^{-\frac{1}{2}} \left(G P^{-\frac{1}{2}} \right)^{\dagger} \begin{bmatrix} x_0 \\ y_f \end{bmatrix} \\ &= P^{-1} G^{\mathsf{T}} \left(G P^{-1} G^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_0 \\ y_f \end{bmatrix}. \end{split}$$

Let y_i denote the *i*th element of vec(Y), it holds

$$\nabla F_{Y,i} = \frac{\partial F(U, X_0, Y)}{\partial y_i}$$
$$= \frac{\partial P^{-1}}{\partial y_i} G^{\mathsf{T}} \left(G P^{-1} G^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_0 \\ y_{\mathsf{f}} \end{bmatrix}$$
(41)
$$+ \frac{P^{-1}}{2} \frac{\partial G^{\mathsf{T}}}{\partial g_i} \left(C P^{-1} C^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_0 \\ y_{\mathsf{f}} \end{bmatrix}$$
(42)

$$+ P^{-1} \frac{\partial}{\partial y_{i}} (GP^{-1}G^{T}) \begin{bmatrix} y_{f} \\ y_{f} \end{bmatrix}$$

$$+ P^{-1}G^{T} \frac{\partial}{\partial y_{i}} \left(\frac{GP^{-1}G^{T}}{\partial y_{i}} \right)^{-1} \begin{bmatrix} x_{0} \\ y_{f} \end{bmatrix}.$$

$$(42)$$

Notice that

$$P = \left(Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger}\right)^{\mathsf{T}} \mathcal{Q} \left(Y \begin{bmatrix} X_0 \\ U \end{bmatrix}^{\dagger}\right) + \begin{bmatrix} 0 & 0 \\ 0 & R \end{bmatrix}$$
$$= \begin{bmatrix} O_T^Y & F_T^Y \end{bmatrix}^{\mathsf{T}} \mathcal{Q} \begin{bmatrix} O_T^Y & F_T^Y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & R \end{bmatrix}, \qquad (44)$$

$$G = \begin{bmatrix} I_n & 0\\ O_F^Y & F_F^Y \end{bmatrix}$$
(45)

since (4) holds by assumption.

Let $\nabla F_{Y,i}^{(1)}$ denote the matrix in (41). This matrix can be written as in (46) (see Box II), where Γ_i is a $nT \times N$ matrix with one entry (corresponding to the element y_i) equal to one and zeros otherwise, and where we used that $\frac{\partial P^{-1}}{\partial y_i} = P^{-1} \frac{\partial P}{\partial y_i} P^{-1}$ (e.g., see Bernstein, 2009). From (46),

$$\begin{aligned} \|\nabla F_{Y,i}^{(1)}\|_{2} \leq \mathcal{E}_{Y,i}^{(1)} \left\| \Gamma_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\dagger} \right\|_{2} \\ \leq \mathcal{E}_{Y,i}^{(1)} \left\| \Gamma_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right)^{-1} \right\|_{2} \\ \leq \mathcal{E}_{Y,i}^{(1)} \left\| \Gamma_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right\|_{2} \sigma_{\min}^{-2} \left(\begin{bmatrix} X_{0} \\ U \end{bmatrix} \right) \\ \leq \mathcal{E}_{Y,i}^{(1)} \left\| \Gamma_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right\|_{2} \frac{1}{cN}, \end{aligned}$$
(49)

where

$$\begin{split} \ell_{Y,1}^{(1)} = & 2 \|P^{-1}\|_{2}^{2} \|Q\|_{2} \| \left[O_{T}^{Y} - F_{T}^{Y} \right] \|_{2} \cdot \\ & \cdot \left\| P^{-1} G^{\mathsf{T}} \left(G P^{-1} G^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_{0} \\ y_{\mathsf{f}} \end{bmatrix} \right\|_{2} \end{split}$$

does not depend on N because of (44), (45). In the first step of (49) we used the submultiplicativity of matrix 2-norm, in the second step the fact that $A^{\dagger} = A^{\mathsf{T}}(AA^{\mathsf{T}})^{-1}$ when A is full-row rank, the third step follows from the fact that $||A^{-1}||_2$ equals the reciprocal of the minimum eigenvalue of A if A is positive definite, and the fourth step from the assumption on $\sigma_{\min}^2([X_0^{\mathsf{T}} \ U^{\mathsf{T}}]^{\mathsf{T}}) \ge cN$. Finally, since the matrix $\Gamma_i \left[X_0^{\mathsf{T}} \ U^{\mathsf{T}}\right]$ has only one row different from zero and the entries of such row independent of N by assumption, (49) implies that $||\nabla F_{Y,i}^{(1)}||_2 \le k_{Y,i}^{(1)}/N$, where $k_{Y,i}^{(1)} > 0$ is a constant independent of N. Next, let $\nabla F_{Y,i}^{(2)}$ denote the matrix in (42). We can write $\nabla F_{Y,i}^{(2)}$ as in

Next, let $\nabla F_{Y,i}^{(2)}$ denote the matrix in (42). We can write $\nabla F_{Y,i}^{(2)}$ as in (47) (see Box II), where Ξ_i is a matrix with one entry set to one and all other entries set to zero, if y_i corresponds to an entry of Y_F , and the zero matrix, otherwise. Similarly as before, we have

$$\|\nabla F_{Y,i}^{(2)}\|_{2} \le \ell_{Y,i}^{(2)} \left\|\Xi_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{T} \right\|_{2} \frac{1}{cN},$$
(50)

where

does not depend on *N*. Since the matrix $\Xi_i \begin{bmatrix} X_0^{\mathsf{T}} & U^{\mathsf{T}} \end{bmatrix}$ is either the zero matrix or has only one row different from zero and the entries of such row are independent of *N* by assumption, (50) implies that $\|\nabla F_{Y,i}^{(2)}\|_2 \leq k_{Y,i}^{(2)}/N$, where $k_{Y,i}^{(2)} > 0$ is a constant independent of *N*.

Finally, let $\nabla F_{Y,i}^{(3)}$ denote the matrix in (43). We can write $\nabla F_{Y,i}^{(3)}$ as in (48) (see Box II). From the triangle inequality of the 2-norm and along the same lines that led to the upper bounds on $\|\nabla F_{Y,i}^{(1)}\|_2$ and $\|\nabla F_{Y,i}^{(2)}\|_2$,

$$\|\nabla F_{Y,i}^{(3)}\|_{2} \leq \ell_{Y,i}^{(3,1)} \left\| \Gamma_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right\|_{2} \frac{1}{cN} + \ell_{Y,i}^{(3,2)} \left\| \Xi_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right\|_{2} \frac{1}{cN} \leq k_{Y,i}^{(3)}/N$$

$$(51)$$

for suitable positive constants $\ell_{Y,i}^{(3,1)}$, $\ell_{Y,i}^{(3,2)}$, $k_{Y,i}^{(3)}$ independent of N.

To conclude, from the triangle inequality and the above upper bounds on $\|\nabla F_{Y,i}^{(1)}\|_2$, $\|\nabla F_{Y,i}^{(2)}\|_2$, $\|\nabla F_{Y,i}^{(3)}\|_2$,

$$\|\nabla F_{y,i}\|_2 \le \|\nabla F_{Y,i}^{(1)}\|_2 + \|\nabla F_{Y,i}^{(2)}\|_2 + \|\nabla F_{Y,i}^{(3)}\|_2 \le \frac{k_{Y,i}}{N}$$

where $k_{Y,i} > 0$ is independent of N .

$$\nabla F_{Y,i}^{(1)} = \frac{\partial P^{-1}}{\partial y_{i}} G^{\mathsf{T}} \left(GP^{-1}G^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_{0} \\ y_{f} \end{bmatrix} = P^{-1} \frac{\partial P}{\partial y_{i}} P^{-1}G^{\mathsf{T}} \left(GP^{-1}G^{\mathsf{T}} \right)^{-1} \begin{bmatrix} x_{0} \\ y_{f} \end{bmatrix}^{\mathsf{T}} QY \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\dagger} + \left(Y \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\dagger} \right)^{\mathsf{T}} QP_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\mathsf{T}} \right)^{\mathsf{T}} QP_{i} \begin{bmatrix} X_{0} \\ U \end{bmatrix}^{\dagger} \right)^{\mathsf{T}} QP_{i} \begin{bmatrix} X_{0}$$

Box II.

Proof of Theorem 4.3. By definition of Δy_f ,

$$\Delta y_{f} = \|\tilde{y}_{f} - y_{f}\|_{2} = \left\|\sum_{i \in \text{supp}(\Delta_{Y})} \delta_{Y,i} \left[O_{T}^{Y} - F_{T}^{Y}\right] \nabla F_{Y,i}\right\|_{2},$$

$$\leq \sum_{i \in \text{supp}(\Delta_{Y})} |\delta_{Y,i}| \left\| \left[O_{T}^{Y} - F_{T}^{Y}\right] \nabla F_{Y,i}\right\|_{2} = \overline{\Delta y_{f}}.$$
(52)

By the monotonicity of probability measures, for any $\tau > 0$, the set inclusion $\{\Delta y_{\rm f} \geq \tau\} \subseteq \{\overline{\Delta y_{\rm f}} \geq \tau\}$ holds, which implies $\mathbb{P}[\Delta y_{\rm f} \geq \tau] \leq \mathbb{P}\left[\overline{\Delta y_{\rm f}} \geq \tau\right]$. Note that $|\delta_{Y,i}|$ are non-negative random variables. Thus, by Markov's inequality (Van der Vaart, 2000) and the linearity of the expected value, for any $\tau > 0$,

$$\mathbb{P}\left[\Delta y_{f} \geq \tau\right] \leq \mathbb{P}\left[\overline{\Delta y_{f}} \geq \tau\right]$$

$$\leq \frac{1}{\tau} \left(\sum_{i \in \text{supp}(\Delta_{Y})} \left\| \begin{bmatrix} O_{T}^{Y} & F_{T}^{Y} \end{bmatrix} \nabla F_{Y,i} \right\|_{2} \mathbb{E}[|\delta_{Y,i}|] \right)$$

$$\leq \frac{c}{\tau} \left(\sum_{i \in \text{supp}(\Delta_{Y})} \left\| \nabla F_{Y,i} \right\|_{2} \mathbb{E}[|\delta_{Y,i}|] \right)$$

$$\leq \frac{c}{\tau} |\operatorname{supp}(\Delta_{Y})| \max_{i} \{ \| \nabla F_{Y,i} \|_{2} \} \max_{i} \{ \mathbb{E}[|\delta_{Y,i}|] \}$$
(53)

where $c = \| \begin{bmatrix} O_T^Y & F_T^Y \end{bmatrix} \|_2$ and $| \operatorname{supp}(\Delta_Y) |$ stands for the cardinality of $\operatorname{supp}(\Delta_Y)$. Since the distributions of $\delta_{X,i}$ are independent of N so are $\mathbb{E}[|\delta_{X,i}|]$. Hence, by (53) and Lemma 4.2, it follows that

$$\mathbb{P}\left[\Delta y_{\mathrm{f}} \geq \tau\right] \leq \frac{k_{Y}}{\tau} \frac{|\operatorname{supp}(\Delta_{Y})|}{N}$$

where $k_Y > 0$ is a constant independent of *N*. This concludes the proof.

A.4. Proof of the results in Section 5

Proof of Theorem 5.1. First, we provide a data-driven solution to the problem in (28). Following a procedure similar to the proof of Theorem 3.1, Problem (28) can be written as in (10), with

$$\gamma = \beta, \ L = \begin{bmatrix} Q^{1/2}X \\ R^{1/2}U \end{bmatrix}, \ W = X_0 K, \text{ and } z = x_0$$

Then, the optimal input and state trajectories of (28) for a given x_0 can be computed as

$$\begin{split} \mathbf{u}_T^* &= UK(I_N - K_W(LK_W)^{\dagger}L)W^{\dagger}x_0, \\ \mathbf{x}_T^* &= XK(I_N - K_W(LK_W)^{\dagger}L)W^{\dagger}x_0. \end{split}$$

From (29) we know that the time-varying controller that solves Problem in (28) satisfies $u(t) = K_{LQR}^t x(t)$, and it does not depend on the initial condition. Let \mathbf{U}_t and \mathbf{X}_t be as in (31). Notice that the *i*th column of \mathbf{X}_t equals the state at time *t* of the optimal state trajectory for (28) with initial state given by the *i*th column of the identity matrix. Then, $\mathbf{U}_t = K_{LQR}^t \mathbf{X}_t$. To conclude, we show that if *A* is invertible then \mathbf{X}_t is invertible. To this end, let $A_t = A + BK_{LQR}^t$ denote the (time-varying) state matrix of the closed-loop system and notice that

$$A_{t} = \underbrace{(I - B(R_{t} + B^{\mathsf{T}}P_{t+1}B)^{-1}B^{\mathsf{T}}P_{t+1})}_{H_{t}}A,$$
(54)

where $P_t \ge 0$ satisfies the Riccati equation (30). Using the push-through identity $(I + XY)^{-1}X = Y(I + YX)^{-1}$, which holds for any matrices X, Y such that (I + XY) is invertible (Bernstein, 2009, Fact 2.16.16), H_t can be written as

$$\begin{aligned} H_t &= I - B(R_t + B^{\mathsf{T}} P_{t+1} B)^{-1} B^{\mathsf{T}} P_{t+1} \\ &= I - BR_t^{-1} (I + B^{\mathsf{T}} P_{t+1} B R_t^{-1})^{-1} B^{\mathsf{T}} P_{t+1} \\ &= I - BR_t^{-1} B^{\mathsf{T}} P_{t+1} (I + BR_t^{-1} B^{\mathsf{T}} P_{t+1})^{-1}. \end{aligned}$$

From the last identity it follows that

$$H_t(I + BR_t^{-1}B^{\mathsf{T}}P_{t+1}) = I,$$

which implies that H_t is invertible with inverse $H_t^{-1} = I + BR_t^{-1}B^T P_{t+1}$. Finally, observe that, for $t \ge 1$,

$$\mathbf{X}_t = A_{t-1}A_t \cdots A_0 = H_{t-1}AH_{t-2}A \cdots H_0A$$

Since the product of invertible matrices is invertible, it follows that if *A* is invertible then X_t is invertible.

References

Aangenent, W., Kostic, D., de Jager, B., van de Molengraft, R., & Steinbuch, M. (2005). Data-based optimal control. In American control conference (pp. 1460–1465). IEEE.

- Abbasi-Yadkori, Y., Lazic, N., & Szepesvári, C. (2019). Model-free linear quadratic control via reduction to expert prediction. In *The 22nd international conference on artificial intelligence and statistics* (pp. 3108–3117). PMLR.
- Al Makdah, A. A., Krishnan, V., Katewa, V., & Pasqualetti, F. (2022). Behavioral feedback for optimal LQG control. In *IEEE conf. on decision and control* (pp. 4660–4666). Cancún, Mexico.
- Al Makdah, A. A., & Pasqualetti, F. (2023). On the sample complexity of the linear quadratic gaussian regulator. In *IEEE conf. on decision and control, Marina Bay Sands, Singapore*. Accepted for publication.
- Anguluri, R., Al Makdah, A. A., Katewa, V., & Pasqualetti, F. (2020). On the robustness of data-driven controllers for linear systems. In *Proceedings of machine learning research: vol. 120, Learning for dynamics & control* (pp. 404–412). San Francisco, CA, USA.
- Åström, K. J., & Murray, R. M. (2010). Feedback systems: An introduction for scientists and engineers. Princeton University Press.
- Baggio, G., Bassett, D. S., & Pasqualetti, F. (2021). Data-driven control of complex networks. *Nature Communications*, 12(1429).
- Baggio, G., Katewa, V., & Pasqualetti, F. (2019). Data-driven minimum-energy controls for linear systems. *IEEE Control Systems Letters*, 3(3), 589–594.
- Baggio, G., & Pasqualetti, F. (2020). Learning minimum-energy controls from heterogeneous data. In *American control conference*. CO, USA: Denver.
- Basile, G., & Marro, G. (1991). Controlled and conditioned invariants in linear system theory. Prentice Hall.
- Berberich, J., Köhler, J., Müller, M. A., & Allgöwer, F. (2020). Data-driven model predictive control with stability and robustness guarantees. *IEEE Transactions on Automatic Control*, 66(4), 1702–1717.
- Berberich, J., Scherer, C. W., & Allgöwer, F. (2022). Combining prior knowledge and data for robust controller design. *IEEE Transactions on Automatic Control.*
- Bernstein, D. S. (2009). Matrix mathematics (2nd ed.). Princeton University Press.
- Bianchin, G. (2023). Data-driven exact pole placement for linear systems. arXiv preprint arXiv:2303.11469.
- Bisoffi, A., De Persis, C., & Tesi, P. (2021). Trade-offs in learning controllers from noisy data. Systems & Control Letters, 154, Article 104985.
- Bisoffi, A., De Persis, C., & Tesi, P. (2022). Data-driven control via Petersen's lemma. Automatica, 145, Article 110537.
- Bradtke, S. J., Ydstie, B. E., & Barto, A. G. (1994). Adaptive linear quadratic control using policy iteration. In *American control conference, Vol. 3* (pp. 3475–3479). IEEE. Breschi, V., Chiuso, A., & Formentin, S. (2023). Data-driven predictive control in a
- stochastic setting: A unified framework. *Automatica*, 152, Article 110961.
- Celi, F., Baggio, G., & Pasqualetti, F. (2022). Closed-form estimates of the LQR gain from finite data. In *IEEE conf. on decision and control* (pp. 4016–4021). Cancún, Mexico.
- Celi, F., Baggio, G., & Pasqualetti, F. (2023a). Data-driven eigenstructure assignment for sparse feedback design. In *IEEE conf. on decision and control, Marina Bay Sands, Singapore.* Accepted for publication.
- Celi, F., Baggio, G., & Pasqualetti, F. (2023b). Distributed data-driven control of network systems. IEEE Open Journal of Control Systems, 93–107.
- Celi, F., & Pasqualetti, F. (2022). Data-driven meets geometric control: Zero dynamics, subspace stabilization, and malicious attacks. *IEEE Control Systems Letters*, 6, 2569–2574.
- Coulson, J., Lygeros, J., & Dörfler, F. (2019). Data-enabled predictive control: In the shallows of the deepc. In *European control conference* (pp. 307–312). Naples, Italy. Coulson, J., Lygeros, J., & Dörfler, F. (2021). Distributionally robust chance constrained
- data-enabled predictive control. IEEE Transactions on Automatic Control, 67(7), 3289–3304.
- da Silva, G. R. G., Bazanella, A. S., Lorenzini, C., & Campestrini, L. (2018). Data-driven LQR control design. *IEEE Control Systems Letters*, 3(1), 180–185.
- De Persis, C., & Tesi, P. (2020). Formulas for data-driven control: Stabilization, optimality and robustness. *IEEE Transactions on Automatic Control*, 65(3), 909–924. De Persis, C., & Tesi, P. (2021). Low-complexity learning of linear quadratic regulators
- from noisy data. Automatica, 128, Article 109548. Dean, S., Mania, H., Matni, N., Recht, B., & Tu, S. (2020). On the sample complexity
- of the linear quadratic regulator. Foundations of Computational Mathematics, 20(4), 633–679.

- Dörfler, F., Tesi, P., & De Persis, C. (2022). On the role of regularization in direct data-driven LQR control. In *IEEE conf. on decision and control* (pp. 1091–1098). Cancún, Mexico: IEEE.
- Dörfler, F., Tesi, P., & De Persis, C. (2023). On the certainty-equivalence approach to direct data-driven LQR design. *IEEE Transactions on Automatic Control*.
- Fazel, M., Ge, R., Kakade, S., & Mesbahi, M. (2018). Global convergence of policy gradient methods for the linear quadratic regulator. In *International conference on machine learning* (pp. 1467–1476). Stockholm, Sweden.
- Gravell, B., Esfahani, P. M., & Summers, T. (2020). Learning optimal controllers for linear systems with multiplicative noise via policy gradient. *IEEE Transactions on Automatic Control*, 66(11), 5283–5298.
- Kailath, T. (1980). Linear systems. Prentice-Hall.
- Krishnan, V., & Pasqualetti, F. (2021). On direct vs indirect data-driven predictive control. In *IEEE conf. on decision and control* (pp. 736–741). Austin, USA.
- Lancaster, P., & Rodman, L. (1995). Algebraic Riccati equations. Clarendon Press.
- Lopez, V. G., & Müller, M. A. (2022). On a continuous-time version of Willems' lemma. In 2022 IEEE 61st conference on decision and control (CDC) (pp. 2759–2764). IEEE.
- Markovsky, I., & Dörfler, F. (2021). Behavioral systems theory in data-driven analysis, signal processing, and control. Annual Reviews in Control, 52, 42–64.
- Mohammadi, H., Zare, A., Soltanolkotabi, M., & Jovanović, M. R. (2019). Global exponential convergence of gradient methods over the nonconvex landscape of the linear quadratic regulator. In *IEEE conf. on decision and control* (pp. 7474–7479). Nice, France.
- Ng, A., & Jordan, M. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Advances in neural information processing systems, Vol. 14.
- Pasqualetti, F., Zampieri, S., & Bullo, F. (2014). Controllability metrics, limitations and algorithms for complex networks. *IEEE Transactions on Control of Network Systems*, 1(1), 40–52.
- Pellegrino, F. A., Blanchini, F., Fenu, G., & Salvato, E. (2023a). Closed-loop control from data driven open-loop optimal control trajectories. In *European control conference* (pp. 1379–1384).
- Pellegrino, F. A., Blanchini, F., Fenu, G., & Salvato, E. (2023b). Data-driven dynamic relatively optimal control. *European Journal of Control*, Article 100839.
- Recht, B. (2019). A tour of reinforcement learning: The view from continuous control. Annual Review of Control, Robotics, and Autonomous Systems, 2, 253–279.
- Rotulo, M., De Persis, C., & Tesi, P. (2020). Data-driven linear quadratic regulation via semidefinite programming. *IFAC-PapersOnLine*, 53(2), 3995–4000.
- Schmitz, P., Faulwasser, T., & Worthmann, K. (2022). Willems' fundamental lemma for linear descriptor systems and its use for data-driven output-feedback MPC. *IEEE Control Systems Letters*, 6, 2443–2448.
- Tu, S., Frostig, R., & Soltanolkotabi, M. (2022). Learning from many trajectories. arXiv preprint arXiv:2203.17193.
- Van der Vaart, A. W. (2000). Cambridge series in statistical and probabilistic mathematics: vol. 3, Asymptotic statistics. Cambridge University Press.
- van Waarde, H. J., Camlibel, M. K., & Mesbahi, M. (2020). From noisy data to feedback controllers: Nonconservative design via a matrix S-lemma. *IEEE Transactions on Automatic Control*, 67(1), 162–175.
- van Waarde, H. J., De Persis, C., Camlibel, M. K., & Tesi, P. (2020). Willems' fundamental lemma for state-space systems and its extension to multiple datasets. *IEEE Control Systems Letters*, 4(3), 602–607.
- Verhoek, C., Tóth, R., Haesaert, S., & Koch, A. (2021). Fundamental lemma for datadriven analysis of linear parameter-varying systems. In 2021 60th IEEE conference on decision and control (CDC) (pp. 5040–5046). IEEE.
- Wang, P., Man, Z., Cao, Z., Zheng, J., & Zhao, Y. (2016). Dynamics modelling and linear control of quadcopter. In 2016 international conference on advanced mechatronic systems (ICAMechS) (pp. 498–503).
- Willems, J. C., & Polderman, J. W. (1997). Introduction to mathematical systems theory: A behavioral approach, Vol. 26. Springer Science & Business Media.
- Willems, J. C., Rapisarda, P., Markovsky, I., & De Moor, B. L. M. (2005). A note on persistency of excitation. Systems & Control Letters, 54(4), 325–329.
- Yu, Y., Talebi, S., van Waarde, H. J., Topcu, U., Mesbahi, M., & Açıkmeşe, B. (2021). On controllability and persistency of excitation in data-driven control: Extensions of willems' fundamental lemma. In *IEEE conf. on decision and control* (pp. 6485–6490). IEEE.