Data-driven predictive control in a stochastic setting: a unified framework☆

Valentina Breschi [a], Alessandro Chiuso [b], Simone Formentin [a],*

[a] *Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, P.za L. Da Vinci, 32, 20133 Milano, Italy*
[b] *Department of Information Engineering, University of Padova, Via Gradenigo 6/b, 35131 Padova, Italy*

ABSTRACT

Data-driven predictive control (DDPC) has been recently proposed as an effective alternative to traditional model-predictive control (MPC) for its unique features of being time-efficient and unbiased with respect to the oracle solution. Nonetheless, it has also been observed that noise may strongly jeopardize the final closed-loop performance, since it affects both the data-based system representation and the control update computed from the online measurements. Recent studies have shown that regularization is potentially a successful tool to counteract the effect of noise. At the same time, regularization requires the tuning of a set of penalty terms, whose choice might be practically difficult without closed-loop experiments In this paper, by means of subspace identification tools, we pursue a three-fold goal: (*i*) we set up a unified framework for the existing regularized data-driven predictive control schemes for stochastic systems; (*ii*) we introduce γ-DDPC, an efficient two-stage scheme that splits the optimization problem in two parts: fitting the initial conditions and optimizing the future performance, while guaranteeing constraint satisfaction; (*iii*) we discuss the role of regularization for data-driven predictive control, providing new insight on *when* and *how* it should be applied. A benchmark numerical case study finally illustrates the performance of γ-DDPC, showing how controller design can be simplified in terms of tuning effort and computational complexity when benefiting from the insights coming from the subspace identification realm.

1. Introduction

Data-driven control (DDC) refers to the science of learning feedback controllers from data, without first undertaking a full modeling study of the plant to control (Formentin, Van Heusden, & Karimi, 2014). Such a direct mapping of data onto the control action is indeed advisable in real-world problems, as modeling usually takes about 75% of the time devoted to a control project (Gevers, 2005), and accurate modeling for control requires significant time and several (costly) technical expertises, e.g., in the process domain and in the statistical tools for system identification. Additionally, accurate modeling may go well beyond what is strictly necessary for control purposes only,

since often times rather limited knowledge of the system dynamics may be required to achieve the desired control objectives (Hjalmarsson, 2005). Early attempts in the direction of DDC date back to 1942, with the first studies by Ziegler and Nichols about PID auto-tuning (Ziegler, Nichols, et al., 1942). More sophisticated, optimization based, approaches have been derived since then for fixed-order controller tuning, leading to a portfolio of techniques suitable for different problem formulations, see, e.g., Breschi and Formentin (2020), Formentin, Campi, Carè, and Savaresi (2019), Formentin and Karimi (2012) and Karimi and Kammer (2017). However, it is only recently that such a paradigm shift in control design could be extended to more complex control architectures, thanks to the availability of large datasets and unparalleled computing power.

In this context lays the uprising interest in data-driven predictive control (DDPC) solutions, that combine the capability of constraint handling of MPC with the flexibility of a data-driven, nonparametric predictor of the system under control. By relying on the so-called "fundamental lemma" (Willems, Rapisarda, Markovsky, & De Moor, 2005) (or variations of the latter), most of existing DDPC techniques replace model equations with suitable

data-based constraints[1] (see e.g., Berberich, Köhler, Müller, & Allgöwer, 2020; Coulson, Lygeros, & Dörfler, 2019). The transition to this data-based framework may lead to different performance than traditional model-based MPC, because of its unique features. For instance, the sub-optimality gap measuring the control performance with respect to the optimal model-based solution (namely, that obtained using the real model of the system) vanishes with the size of the dataset. Moreover, model-free predictive control may indirectly address the bias/variance trade-off in a more efficient manner. Indeed, it will not incur in the asymptotic bias induced by inaccurate modeling when complexity constraints are imposed on the model structure, as discussed in Krishnan and Pasqualetti (2021).

The shift from a model-based to a data-driven framework is well established in case of purely deterministic systems, whereas many of the attempts made to counteract the effect of noise in the presence of stochastic disturbances lead to approximations that may deteriorate the closed-loop performance. For instance, in Berberich et al. (2020), the authors propose a regularized DDPC scheme, with guarantees of practical exponential stability in closed loop in the presence of *bounded* additive output noise. The key ingredients to achieve this result are two: (*i*) some bounded slack variables to account for the noisy data used for prediction, and (*ii*) some suitable regularization terms. In Alanwar, Stürz, and Johansson (2021), a slightly different scheme is used, which computes the data-driven reachable set based on a matrix zonotope recursion starting from the measured output. For this scheme, the authors show they can guarantee robust constraint satisfaction, again in case of *bounded* process and measurement noises. The case of stochastic (white) measurement noise is addressed in Yin, Iannelli, and Smith (2021), where a maximum likelihood framework is proposed to estimate the data-based constraints aimed to replace the model equations in the MPC formulation. the resulting scheme is an iterative *two-stage* approach, where at each iteration first a model encoded by a data matrix constraint *must* be identified and then the online predictive control is computed. An approach to handle stochastic noise in the direct framework proposed in Berberich et al. (2020) and Coulson et al. (2019) is given in the recent paper (Dörfler, Coulson, & Markovsky, 2022). In this contribution, the authors exploit regularization as the *key tool* to handle the presence of noise in the output measurements, and empirically discuss the performance of different regularization schemes. Approaches for DDPC with regularization are also shown to be distributionally robust in Coulson, Lygeros, and Dörfler (2019). One of the regularized schemes proposed in Dörfler et al. (2022) is then connected with *Subspace Predictive Control* (SPC) (Favoreel, Moor, & Gevers, 1999) by Fiedler and Lucia (2021), where the introduction of additional slacks is further proposed to cope with noisy online data. Kalman filter approaches have finally been suggested in Alpago, Dörfler, and Lygeros (2020) to filter out the effect of noise in the context of DeePC approaches.

In this paper, we consider a stochastic setting where both measurement and process noise are considered. Within this framework, our contribution is three-fold.

C1. By revising foundational results in subspace identification, we show that the seminal regularized DDPC schemes in Berberich et al. (2020), Dörfler et al. (2022) and Fiedler and Lucia (2021) can all be recast into a *unified framework*, stemming from the constrained counterpart of the SPC scheme originally proposed in Favoreel et al. (1999).

C2. Based on this unified framework, we discuss how the choice of *key hyperparameters* in Berberich et al. (2020), Dörfler et al. (2022) and Fiedler and Lucia (2021) can be guided by known results in subspace identification. These insights potentially allow the final user to select the regularization parameters in those schemes with *less* closed-loop tests, while possibly allow one to *avoid* such experiments if the available dataset is large.

C3. We show that the parameterization of the predictor and the control input exploited to solve the DDPC problem can be decomposed in three terms with specific roles. This decomposition allows us to *split* the DDPC problem into two sub-problems of smaller dimensions, respectively devoted to: (*i*) fit the initial conditions embedded in the input/output data streams collected online; (*ii*) optimize performance in prediction, while avoiding constraint violations. The introduction of this two-stage scheme, which we call *γ-DDPC* from now on, allows for a reduction in the computational complexity of the overall DDPC formulation, while providing the final user with a more transparent overview of the main players of the control scheme.

By means of a benchmark numerical example, we show the performance of γ-DDPC and we validate the insights gained from subspace identification about the role of regularization, showing how the latter can be actively exploited to *avoid* (or at least reduce the number of) the closed loop experiments needed to tune the regularization weights through cross-validation.

The remainder of the paper is organized as follows. In Section 2, we formally define the control problem of interest and its data-driven counterpart. Section 3 reviews in detail subspace identification concepts to give a deeper insight into the employed system description, ultimately leading to the constrained SPC formulation at the core of the unified framework for regularized DDPC techniques presented in Section 4. In light of the preceding analysis, Section 5 introduces γ-DDPC and discusses the role of regularization in the data-driven control framework. The benchmark numerical example of Section 6 illustrates the effectiveness of the γ-DDPC perspective in designing a satisfactory control action. The paper is ended by some concluding remarks.

Notation. Matrices will be denoted with capitals (e.g. A), column vectors will be denoted with lowercase letters (e.g. a). The transpose of A will be denoted with $A^\top$; the notation $A^\dagger$ will denote the Moore–Penrose pseudo-inverse of A. Given deterministic (vector) sequences $a(t)$, $b(t)$ the notation $a(t) = O(b(t))$ means that there exist M and $c < \infty$ such that, for all $t > M$,

$$\|a(t)\| \le c \|b(t)\|. \tag{1}$$

Similarly we say that $a(t) = o(b(t))$ if

$$\lim_{t \to \infty} \frac{\|a(t)\|}{\|b(t)\|} = 0,$$

or, equivalently, that for all $\epsilon > 0$, there exists $M < \infty$ such that, for all $t > M$,

$$\|a(t)\| \le \epsilon \|b(t)\|. \tag{2}$$

Probabilistic versions of $O(\cdot)$ and $o(\cdot)$ (i.e., with conditions (1) and (2) holding in probability) will be denoted by $O_P(\cdot)$ and $o_P(\cdot)$, see e.g., van der Vaart (1998). Given a and b, we use the symbols $\doteq$ and $\triangleq$ to denote equality up to $o_P(1/\sqrt{N})$ and up to $O_P(1/\sqrt{N})$, respectively. Namely

$$a \doteq b \iff a = b + o_P(1/\sqrt{N}), \tag{3a}$$

$$a \triangleq b \iff a = b + O_P(1/\sqrt{N}). \tag{3b}$$

---

[1] Such constraints are an implicit, nonparametric, mapping of the input/output relationships. According to this interpretation, some researchers legitimately prefer to denote the strategies described herein as "indirect". For this reason, we will simply talk about *data-driven predictive control* from now on.

$\Pi_A[B]$ denotes the orthogonal projection of the (rows of the) matrix B on the row span of the matrix A, i.e.,

$$\Pi_A[B] = BA^\top (AA^\top)^\dagger A.$$

Similarly $\Pi_{A,C}[B]$ indicates the projection of B onto the row span of A and C. Finally, given a signal $w(k) \in \mathbb{R}^s$, we define the associated Hankel matrix $W_{[t_0,t_1],N} \in \mathbb{R}^{s(t_1-t_0+1)\times N}$ as:

$$W_{[t_0,t_1],N} := \frac{1}{\sqrt{N}} \begin{bmatrix} w(t_0) & w(t_0+1) & \cdots & w(t_0+N-1) \\ w(t_0+1) & w(t_0+2) & \cdots & w(t_0+N) \\ \vdots & \vdots & \ddots & \vdots \\ w(t_1) & w(t_1+1) & \cdots & w(t_1+N-1) \end{bmatrix}, \tag{4}$$

while the shorthand $W_{t_0} := W_{[t_0,t_0],N}$ is used to denote the Hankel containing a single row, namely:

$$W_{t_0} := \frac{1}{\sqrt{N}} \begin{bmatrix} w(t_0) & w(t_0+1) & \cdots & w(t_0+N-1) \end{bmatrix}. \tag{5}$$

2. Setting and goal

Consider an *unknown* discrete-time, *linear time-invariant* (LTI) stochastic plant, whose behavior can always be described by the so-called *innovation-form* equations

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) + Ke(t) \\ y(t) = Cx(t) + Du(t) + e(t), \end{cases} \quad t \in \mathbb{Z} \tag{6}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $e(t) \in \mathbb{R}^p$ are the state, input and innovation process respectively, while $y(t) \in \mathbb{R}^p$ is the corresponding output signal. Without loss of generality we shall assume that (6) is minimal (i.e., reachable and observable).

Given a constant reference signal y_r, a constant reference input u_r, and a control horizon T, the receding horizon predictive control problem can be framed as follows:

$$\underset{u(k),k\in[t,t+T)}{\text{minimize}} \frac{1}{2} \left[\sum_{k=t}^{t+T-1} \mathbb{E}\left[\|y(k)-y_r\|_Q^2\right] + \|u(k)-u_r\|_R^2 \right] \tag{7a}$$

$$\text{s.t. } x(k+1) = Ax(k) + Bu(k) + Ke(k), \; k \in [t,t+T), \tag{7b}$$

$$y(k) = Cx(k) + Du(k) + e(k), \; k \in [t,t+T), \tag{7c}$$

$$x(t) = x_{init}, \tag{7d}$$

$$u(k) \in \mathcal{U}, \; \mathbb{E}[y(k)] \in \mathcal{Y}, \; k \in [t,t+T), \tag{7e}$$

where $k \in \mathbb{Z}$, x_{init} is the state at time t, $e(k)$ is a zero mean noise with variance $Var\{e(k)\}$, the sets $\mathcal{U}, \mathcal{Y}$ denote inputs and output constraints, and the expectation $\mathbb{E}[\cdot]$ is taken w.r.t. the future noise sequence $e(k)$, $k \in [t,t+T)$, and *conditionally* on the initial state x_{init} and the future input trajectory $u_f := \{u(k), k \in [t,t+T)\}$. The tunable symmetric weights $Q \in \mathbb{R}^{p\times p}$ and $R \in \mathbb{R}^{m\times m}$, with $Q \succeq 0$ and $R \succ 0$, have to be selected to trade-off between tracking performance and the required control effort. Our goal is to solve problem (7) when the systems matrices A, B, C, D, K are *not known* and only a sequence of input output data $\mathcal{D}_{N_{data}} = \{u(j), y(j)\}_{j=1}^{N_{data}}$ collected in open loop[2] from system (6) is available.

2.1. Features of the predictive control problem

We now elaborate on the optimization problem (7) and make two important observations:

1. Problem (7) can be equivalently formulated only in terms of the so called "deterministic" part of the stochastic system (6), i.e., the one depending only on the control input and the initial state, but not on the noise $e(k)$.
2. The initial state x_{init} at time t does not have to be available. Indeed, it can be accounted for with arbitrary accuracy based on a sufficiently long window of past input–output observations.

To show that the first point holds, it is useful to rewrite the control problem (7) exploiting the decomposition of second order moments as the sum of squared means plus variance, i.e.,

$$\mathbb{E}\left[\|y(k)-y_r\|_Q^2\right] = \|\mathbb{E}[y(k)]-y_r\|_Q^2 + \underbrace{\mathbb{E}\left[\|y(k)-\mathbb{E}[y(k)]\|_Q^2\right]}_{\text{independent of } u(k)}$$

Since the variance term $\mathbb{E}[\|y(k)-\mathbb{E}[y(k)]\|_Q^2]$ is independent of the input signal $u(k)$, $k \in [t,t+T)$, only the conditional (given x_{ini} and u_f) mean value of the output, namely $y^d(k) := \mathbb{E}[y(k)]$ affects the optimization problem. Denoting with $x^d(k)$ the conditional mean of $x(k)$, i.e. $x^d(k) := \mathbb{E}[x(k)]$, it is straightforward to see that the optimal control problem (7) can be equivalently recast as

$$\underset{u(k),k\in[t,t+T)}{\text{minimize}} \frac{1}{2} \left[\sum_{k=t}^{t+T-1} \|y^d(k)-y_r\|_Q^2 + \|u(k)-u_r\|_R^2 \right] \tag{8a}$$

$$\text{s.t. } x^d(k+1) = Ax^d(k) + Bu(k), \; k \in [t,t+T), \tag{8b}$$

$$y^d(k) = Cx^d(k) + Du(k), \; k \in [t,t+T), \tag{8c}$$

$$x^d(t) = x_{init}, \tag{8d}$$

$$u(k) \in \mathcal{U}, \; y^d(k) \in \mathcal{Y}, \; k \in [t,t+T). \tag{8e}$$

Even though only the "deterministic" part of the system influences the optimal control problem, it is important to stress that measured data are indeed affected by noise. This should be accounted for when exploiting measured data $\mathcal{D}_{N_{data}}$ to solve (8).

As it concerns the second observation, to prove its validity we exploit the fact that (6) can be written in innovation (or "whitening" Chiuso & Picci, 2005) form. Accordingly, it holds that

$$\begin{cases} x(k+1) = (A-KC)x(k) + Bu(k) + Ky(k) \\ e(k) = y(k) - Cx(k) + Du(k), \end{cases} \tag{9}$$

and, for any $\rho > 0$, $\rho \in \mathbb{Z}$,

$$x(t) = (A-KC)^\rho x(t-\rho) + \sum_{p=1}^{\rho} \left[\Phi_p u(t-p) + \Psi_p y(t-p) \right], \tag{10}$$

where $\Phi_p = (A-KC)^{p-1}B$ and $\Psi_p = (A-KC)^{p-1}K$. By denoting with λ_{max} the eigenvalues of $A-KC$ of largest absolute value, under the (mild) assumption that the matrix $A-KC$ is strictly stable, i.e., $|\lambda_{max}| < 1$, we have that:

$$x(t) = \mathcal{C} \begin{bmatrix} u_t^- \\ y_t^- \end{bmatrix} + \underbrace{O(|\lambda_{max}|^\rho)}_{\to 0 \text{ for } \rho\to\infty} \tag{11}$$

where the $O(\cdot)$ term goes to zero exponentially; $\mathcal{C}$ stacks the (reversed) controllability matrices $\mathcal{C}_u$ and $\mathcal{C}_y$, i.e.,

$$\mathcal{C} = \begin{bmatrix} \mathcal{C}_u & \mathcal{C}_y \end{bmatrix} = \begin{bmatrix} \Phi_\rho & \cdots & \Phi_2 & \Phi_1 & \Psi_\rho & \cdots & \Psi_2 & \Psi_1 \end{bmatrix},$$

and

$$u_t^- := \begin{bmatrix} u(t-\rho) \\ \vdots \\ u(t-2) \\ u(t-1) \end{bmatrix}, \quad y_t^- := \begin{bmatrix} y(t-\rho) \\ \vdots \\ y(t-2) \\ y(t-1) \end{bmatrix} \tag{12}$$

---

[2] Extension to data collected in closed-loop is possible. Yet, for the sake of exposition, its treatment is deferred to future publications.

are *noisy* collections of past inputs and outputs. The relation in (11) thus guarantees that, up to $O(|\lambda_{max}|^\rho)$ terms, the initial state can be uniquely reconstructed with a finite window of past data.

Remark 1 (*State/Data Relation*). In the so-called "deterministic case", i.e., when there is no process/measurement noise in (6), the state at time t is a (deterministic) function of a *finite* past window of input–output data, As such, $\exists \mathcal{C}_{det}$ such that

$$x(t) = \mathcal{C}_{det} \begin{bmatrix} u_t^- \\ y_t^- \end{bmatrix},$$

provided $\rho \geq n$. This is a trivial consequence of (10) and of the observability of the system.

Remark 2 (*Choice of ρ - Part I*). In (subspace) system identification, see e.g., Bauer (2005) and Chiuso (2007a, 2007b, 2010) the quantity ρ, known also as the "past horizon", has to be determined from measured data trading off bias and variance. Indeed, ρ should be large, so that the quantity $O(|\lambda_{max}|^\rho)$ can be neglected, but a large ρ ultimately requires estimating larger sample covariance matrices. A simple and effective way of determining ρ in a data-driven fashion is by using Akaike's criterion (e.g., FPE) (Akaike, 1969), with the latter choice also guaranteeing that $\|(A - KC)^\rho\| = O(|\lambda_{max}|^\rho) = o_P(1/\sqrt{N_{data}})$. This is in contrast with common practice in the literature of DDPC where the length ρ of the past horizon is not linked to the eigenvalues of $(A - KC)$, but rather it is generally chosen based on (e.g., an upper bound of) the "order" n of the deterministic model. Finally, note that $(A-KC)$ encodes information both on the deterministic dynamics and the noise properties. Hence, the choice of ρ is intimately related to the stochastic nature of the disturbances, a feature commonly neglected in DDPC schemes.

3. DDPC formulation via subspace methods

In this Section, we exploit ideas from subspace identification to recast Problem (8) in terms of observed input output data $D_{N_{data}}$. The results in this section are standard in subspace identification and can be found in several references, see for instance (Bauer, 2005; Bauer & Jansson, 2000; Chiuso, 2006, 2007a, 2007b; Dahlén & Scherrer, 2004).

Let us first define the joint input/output process

$$z(k) := \begin{bmatrix} u(k) \\ y(k) \end{bmatrix},$$

and introduce the shorthands for the "past" Hankel matrices, namely,

$$U_P := U_{[0,\rho-1],N}, \; Y_P := Y_{[0,\rho-1],N}, \; Z_P := Z_{[0,\rho-1],N} \quad (13)$$

and the "future" ones, i.e.,

$$U_F := U_{[\rho,\rho+T-1],N}, \; Y_F := Y_{[\rho,\rho+T-1],N},$$
$$E_F := E_{[\rho,\rho+T-1],N} \quad (14)$$

Note that, once the lengths of both the "past" ρ and "future" T are fixed, the number of columns N of the Hankel data matrices is chosen in such a way that all the available data are exploited, namely $N := N_{data} - T - \rho$. Let us further introduce the extended observability matrix $\Gamma \in \mathbb{R}^{pT \times n}$ associated with the system in (6), namely

$$\Gamma = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{T-1} \end{bmatrix}, \quad (15)$$

and the Toeplitz matrices $\mathcal{H}_d \in \mathbb{R}^{pT \times mT}$ and $\mathcal{H}_s \in \mathbb{R}^{pT \times pT}$ formed with its Markov parameters, i.e.,

$$\mathcal{H}_d = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{T-2}B & CA^{T-3}B & CA^{T-4}B & \dots & D \end{bmatrix}, \quad (16a)$$

$$\mathcal{H}_s = \begin{bmatrix} I & 0 & 0 & \dots & 0 \\ CK & I & 0 & \dots & 0 \\ CAK & CK & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{T-2}K & CA^{T-3}K & CA^{T-4}K & \dots & I \end{bmatrix}. \quad (16b)$$

Based on (11) and provided ρ is chosen in a data-driven fashion as discussed in Remark 2, X_ρ can be written as

$$X_\rho = \underbrace{\mathcal{C}_u U_P + \mathcal{C}_y Y_P}_{:=\mathcal{C}Z_P} + \underbrace{(A - KC)^\rho X_0}_{O(|\lambda_{max}|^\rho)} \doteq \mathcal{C}Z_P, \quad (17)$$

where $Z_P = \begin{bmatrix} U_P^\top & Y_P^\top \end{bmatrix}^\top$. The Hankel matrix of future outputs Y_F thus satisfies the following:

$$\begin{aligned} Y_F &= \Gamma X_\rho + \mathcal{H}_d U_F + \mathcal{H}_s E_F \\ &\doteq \mathcal{C}Z_P + \mathcal{H}_d U_F + \mathcal{H}_s E_F, \end{aligned} \quad (18)$$

which is the equation often considered as a starting point in subspace identification and control (Chiuso & Picci, 2005; Favoreel et al., 1999; Van Overschee & De Moor, 1994). We can now characterize the future noise E_F according to the following.

Lemma 1 (*Projection of Noise*). *For any fixed ρ in (12), it holds that*

$$\Pi_{Z_P, U_F}(E_F) = \Upsilon \begin{bmatrix} Z_P \\ U_F \end{bmatrix}, \quad (19)$$

where $\|\Upsilon\| = O_P\left(\frac{1}{\sqrt{N}}\right)$ and Z_P, U_F are defined as in (13) and (14).

Proof. By definition,

$$\Pi_{Z_P, U_F}(E_F) = \underbrace{E_F \begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix} \left[\begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix} \right]^{-1}}_{\Upsilon} \begin{bmatrix} Z_P \\ U_F \end{bmatrix},$$

so that

$$\Upsilon = \underbrace{E_F \begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix}}_{[\hat{\Sigma}_{e_F z_P} \; \hat{\Sigma}_{e_F u_F}]} \left[\begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix} \right]^{-1}.$$

It is sufficient to observe that the term on the left-hand side $\hat{\Sigma}_{e_F z_P}$ and $\hat{\Sigma}_{e_F u_P}$ are sample cross-covariances between future innovations and past data (z_P) or future inputs u_F, and thus converge to zero in probability with rate $\frac{1}{\sqrt{N}}$, whereas the rightmost term converges to the input–output covariance matrix, which is bounded away from zero thanks to the persistency of excitation assumptions, see, e.g., Lemma 3 later.

This Lemma further allows us to characterize the future outputs Y_F as follows.

Lemma 2 (*Projection of the Output*). *The projection $\hat{Y}_F := \Pi_{Z_P, U_F}(Y_F)$ satisfies*

$$\begin{aligned} \hat{Y}_F &= \Gamma \hat{X}_\rho + \mathcal{H}_d U_F + \mathcal{H}_s \Pi_{Z_P, U_F}(E_F) \\ &\triangleq \Gamma \mathcal{C}Z_P + \mathcal{H}_d U_F, \end{aligned} \quad (20)$$

where $\hat{X}_\rho := \Pi_{Z_P, U_F}(X_\rho) \doteq \mathcal{C}Z_P$.

Proof. The proof straightforwardly follows from the observation that the projection is a linear operator, by exploiting Lemma 1 on the projection of the noise term and Eq. (17) on the approximation of the state using a finite set of past data.

Given the projected initial condition $\hat{X}_\rho$ and the input U_F, Lemma 2 establishes that the projected output $\hat{Y}_F$ equals the evolution of the deterministic part of the system (6), up to $O_P(1/\sqrt{N})$ terms. This result is formalized in the following Theorem.

Theorem 1 (*Output/Data Relation*). *Given any $\alpha \in \mathbb{R}^N$, the vector $\hat{y}_f^d := \hat{Y}_F\alpha$ satisfies the relation*

$$\hat{y}_f^d = \Gamma\hat{x}^d(t) + \mathcal{H}_d u_f + O_P\left(\frac{1}{\sqrt{N}}\right) \tag{21}$$
$$\triangleq \Gamma\hat{x}^d(t) + \mathcal{H}_d u_f,$$

where

$$\hat{x}^d(t) := \hat{X}_\rho\alpha \doteq \mathcal{C}Z_P\alpha = \mathcal{C}z_{init}, \tag{22a}$$

$$u_f := \begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+T-1) \end{bmatrix} = U_F\alpha, \tag{22b}$$

and $z_{init} := Z_P\alpha$.

Proof. The proof is an immediate consequence of Lemma 2. In fact, defining $\hat{y}_f^d := \hat{Y}_F\alpha$ and using Eq. (20), we have that

$$\hat{y}_f^d := \hat{Y}_F\alpha \triangleq \Gamma\mathcal{C}\underbrace{Z_P\alpha}_{:=x_{init}} + \mathcal{H}_d\underbrace{U_F\alpha}_{:=u_f}.$$

The result in Theorem 1 should be read as follows. If the sequence of past input–output $u(k)$ and $y(k)$ for $k \in [t-\rho, t-1]$ equals z_{init} and the future inputs $u(k)$ in the time window $k \in [t, t+T-1]$ (see (22b)) are given by u_f, the corresponding "deterministic" output, i.e.,

$$y_f^d := \begin{bmatrix} y^d(t) \\ y^d(t+1) \\ \vdots \\ y^d(t+T-1) \end{bmatrix},$$

is a linear transformation through α of the projected future outputs $\hat{Y}_F$, up to $O_P(1/\sqrt{N})$ terms.

Towards DDPC

For every pair of initial conditions and future inputs that can be written as linear combinations of Z_P and U_F (see (13) and (14)), Theorem 1 shows that one can compute the deterministic output of (6) (up to $O_P(1/\sqrt{N})$ terms) from a *finite set* input–output data only, *without knowing* the true system (6). Under the additional assumption that the training input $u(t)$ has a full rank spectral density matrix and the innovation process has positive definite variance $Var\{e(t)\} > 0$,[3] we can guarantee that the matrices Z_P and U_F have full rank, so that *any* possible initial condition and sequence of control inputs can be generated by linear combination of their columns. The following lemma formalizes this result.

---

[3] Since the our purpose *is not* to discuss the weakest conditions under which the results of Theorem 1 can be generalized, here we make a sufficient assumption that is general enough for being widely applicable in practice.

Lemma 3 (*Persistency of Excitation*). *If the input process has full rank spectral density that is bounded away from zero and $Var\{e(t)\} > 0$, then for any choice of ρ and T and provided $N > (m+p)(\rho+T)$, the block Hankel matrix*

$$Z_{data} := \begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} \in \mathbb{R}^{(m+p)(\rho+T) \times N} \tag{23}$$

has full rank almost surely.

Proof. The proof is a direct consequence of the fact that, under the stated assumptions, the joint spectral density matrix of the input–output process $z(t) := [u^\top(t)\ y^\top(t)]^\top$ does not vanish on the unit circle and, therefore, the intersection between the (joint) past and input spaces contains only the zero random variable (see e.g. Hannan and Poskitt (1988)). Thus, the Hankel matrix formed with input output trajectories has full rank almost surely.

Under the latter, the result in Theorem 1 can be generalized to all initial conditions and future inputs, as stated in the main result of this Section.

Theorem 2 (*Output/Data Relation - Generalized*). *Under the assumptions in Lemma 3, given any (past) joint input and output trajectory*

$$z_{init} := \begin{bmatrix} z(t-\rho) \\ \vdots \\ z(t-2) \\ z(t-1) \end{bmatrix}, \tag{24}$$

and any choice of the future control input

$$u_f := \begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+T-1) \end{bmatrix}, \tag{25}$$

the corresponding "deterministic" output

$$y_f^d := \begin{bmatrix} y^d(t) \\ y^d(t+1) \\ \vdots \\ y^d(t+T-1) \end{bmatrix}$$

satisfies:

$$y_f^d = \hat{Y}_F\alpha^\star + O_P(1/\sqrt{N}) \triangleq \hat{Y}_F\alpha^\star \tag{26}$$

where $\alpha^\star$ is the minimum-norm solution of the system of linear equations:

$$\begin{bmatrix} z_{init} \\ u_f \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \end{bmatrix}\alpha \tag{27}$$

where $\hat{Y}_F := \Pi_{Z_P, U_F}(Y_F)$.

Proof. Under the assumption of Lemma 3, the matrix Z_{data} has full rank and, therefore, $\forall z_{init}$ and u_f, there exists α such that

$$\begin{bmatrix} z_{init} \\ u_f \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \end{bmatrix}\alpha. \tag{28}$$

Thus, exploiting Theorem 1, the corresponding deterministic output satisfies

$$y_f^d = \hat{Y}_F\alpha + O_P(1/\sqrt{N}) \triangleq \hat{Y}_F\alpha.$$

This is true for *all* possible solutions of (28), and in particular it holds for its minimum-norm solution α^*.

Remark 3 (*The Case of Deterministic Systems*). The reader may observe that, when $e(t) = 0$ (that is the system is actually deterministic), Lemma 3 does not hold. Indeed, for $\rho > n$, it is well known (see, e.g., Moonen, De Moor, Vandeberghe, and Vandewalle (1989)) that the Hankel matrix Z_{data} in (23) and $Z_P \in \mathbb{R}^{(m+p)\rho \times N}$ in (27) have rank equal to

$$\text{rank}(Z_{data}) = n + m(\rho + T) < (m + p)(\rho + T),$$

$$\text{rank}(Z_P) = n + m\rho < (m + p)\rho.$$

These relations are indeed the basis for the so-called "intersection algorithms" in subspace identification, and also can be seen as algebraic formulations of the well known "Willems' fundamental lemma" (Willems et al., 2005). Nonetheless, in this case, any finite (deterministic) trajectory z_{init} of the system (6) belongs to the column span of Z_P. As such, provided that z_{init} is an "admissible" sequence of input/output pairs of the given deterministic system, then (27) has a solution.

Based on the previous results, we are now ready to recast the control problem (8) in a data driven fashion as follows:

$$\underset{u_f}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) \tag{29a}$$

$$\text{s.t.} \quad \alpha^\star = \begin{bmatrix} Z_P \\ U_F \end{bmatrix}^\dagger \begin{bmatrix} z_{init} \\ u_f \end{bmatrix}, \tag{29b}$$

$$y_f^d = \hat{Y}_F \alpha^\star, \tag{29c}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t+T), \tag{29d}$$

where

$$J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) = \frac{1}{2} \left[\sum_{k=t}^{t+T-1} \|y^d(k) - y_r\|_Q^2 + \|u(k) - u_r\|_R^2 \right], \tag{29e}$$

and $\hat{Y}_F := \Pi_{Z_P, U_F}(Y_F)$, while z_{init} and u_f are defined as in (24) and (25). Except for the use of a slightly different notation and the introduction of constraints, the problem in (29) corresponds to the *Subspace Predictive Control* (SPC) problem firstly formalized in Favoreel et al. (1999).

4. A unified outlook on DDPC problems

Recent papers have discussed problems that are very similar to (29), generally starting from a *deterministic* viewpoint, i.e., assuming that $e(t) = 0$, $\forall t$ in (6), and then coping with measurement noise by introducing slack variables and regularization terms. However, by reformulating these problems with our notation, we will show that *all of them can be cast into a unified framework*. In particular, we will show the connections among the problem in (29), the one with slacks on the initial conditions proposed in Fiedler and Lucia (2021, Section IV.B), the formulation tailored to cope with bounded noise introduced in Berberich et al. (2020) and that with elastic net regularization given in Dörfler et al. (2022, Section IV.D).

4.1. SPC with slacks

Based on our notation, the SPC problem tackled in Fiedler and Lucia (2021) to handle non-deterministic scenarios (with measurement noise only) can be recast as follows:

$$\underset{u_f, \sigma \geq 0}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) + \lambda \|\sigma\|_2^2 \tag{30a}$$

$$\text{s.t.} \quad \alpha = \begin{bmatrix} Z_P \\ U_F \end{bmatrix}^\dagger \left(\begin{bmatrix} z_{init} \\ u_f \end{bmatrix} + \begin{bmatrix} \sigma \\ 0 \end{bmatrix} \right), \tag{30b}$$

$$y_f^d = \hat{Y}_F \alpha, \tag{30c}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t+T), \tag{30d}$$

where $\sigma \in \mathbb{R}^{\rho(m+p)}$ is a slack variable to be optimized in order to cope with noise on the data used to build z_{init}, while $\lambda > 0$ is a tunable parameter.[4] The role of the additional slack σ introduced in (30) is linked to the results presented in Section 2.1 by the following proposition.

Lemma 4 (*Asymptotic Regularization with Slacks*). *Assume that the cost $J(\cdot)$ in (30a) is equal to (29e). Then the solution to problem (29) coincides with the one of (30) when $\lambda \to \infty$. Moreover, $\lambda \to \infty$ is the optimal choice even for finite, but large, N_{data} and ρ chosen according to the Akaike's criterion.*

Proof. The proof of the first statement is a direct consequence of the formulations of the problems. The second claim straightforwardly follows from (11) and Remark 2, showing that the error due to finite past is $o_P(1/\sqrt{N_{data}})$ (and thus can be neglected), whereas the error due to finite N_{data} in the projection (26) is instead $O_P(1/\sqrt{N_{data}})$.

This additional insight on problem (30) provides a direct connection between ρ and N_{data} and the value of the slack variables needed to counteract the effect of noise on the initial conditions. Indeed, when ρ is selected according to Akaike's criterion, Remark 2 directly links the accuracy of the reconstructed state with the dimension of the dataset. As such, for large N_{data}, then $\lambda \to +\infty$ is the optimal choice. This will be confirmed by the simulation results reported in Section 6 (see Fig. 8(a)).

4.2. DDPC with bounded measurement noise

Let us now focus on a stochastic settings in which $K = 0$ in (9), the measurement noise $e(t)$ is bounded, namely $\|e(t)\|_\infty \leq \bar{\varepsilon}$, and $\bar{\varepsilon}$ is assumed to be *known*. In our framework, the regularized problem proposed in Berberich et al. (2020) to tackle this scenario can be rewritten as follows:

$$\underset{u_f, y_f^d, \alpha, \sigma}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) + \lambda_\alpha \bar{\varepsilon} \|\alpha\|_2^2 + \lambda_\sigma \|\sigma\|_2^2 \tag{31a}$$

$$\text{s.t.} \quad \begin{bmatrix} z_{init} + \mathbb{1}_y \sigma_{init} \\ u_f \\ y_f^d + \sigma_y \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} \alpha, \tag{31b}$$

$$\begin{bmatrix} u(k) \\ y^d(k) \end{bmatrix} = \begin{bmatrix} u_r \\ y_r \end{bmatrix}, \ k \in [t+T-\rho, t+T), \tag{31c}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t+T), \tag{31d}$$

$$\|\sigma\|_\infty \leq \bar{\varepsilon}(1 + \|\alpha\|_1), \ k \in [t, t+T), \tag{31e}$$

where

$$\sigma = \begin{bmatrix} \sigma_{init} \\ \sigma_y \end{bmatrix} \in \mathbb{R}^{p(T+\rho)}, \tag{31f}$$

is a vector of slacks accounting for the noise acting on the output measurements used to build Z_P and Y_F, while $\mathbb{1}_y$ is a selector function, introduced to add the slack only on the initial outputs comprised in z_{init}. Note that, the constraint in (31c) is a terminal ingredient introduced to guarantee practical stability and recursive feasibility of the DDPC scheme, and the inequality in (31e) is a *non-convex*[5] constraint that connects the slack variables to the known features of the measurement noise.

---

[4] By considering a diagonal matrix Λ rather than a scalar λ, different weights can be chosen for the slack acting on past inputs and outputs, like in the framework proposed in Fiedler and Lucia (2021).

[5] The constraint in (31e) cannot be enforced without resorting to a non-convex optimization routine. Thus, the entity of the slack is practically contained by a proper tuning of λ_σ.

By recasting the robust DDPC formulation in Berberich et al. (2020) within our framework, we now establish its relationship with the constrained SPC scheme in (29) through the following result, which shows that the formulation in Berberich et al. (2020) complemented with an additional constraint on the slack variable σ_y is equivalent to constrained SPC, for a suitable choice of regularization parameters.

Theorem 3 (*Regularization with Bounded Noise*). *Let the cost $J(\cdot)$ in (31) be defined as in (29e) and the non-convex constraint in (31e) be neglected. Assume that the SPC problem in (8) is augmented with the terminal ingredient in (31c). Then, under the additional constraint on the slack variable $\sigma_y = Y_F(I - \Pi)\alpha$, where Π is the orthogonal projector onto the column span of $\begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix}$, i.e.,*

$$\Pi := \begin{bmatrix} Z_P^\top & U_F^\top \end{bmatrix} \begin{bmatrix} Z_P \\ U_F \end{bmatrix}^\dagger, \tag{32}$$

the solutions of (29) with terminal constraints and (31) coincide for $\lambda_\alpha \bar\varepsilon = 0$ and $\lambda_\sigma \to \infty$.

Proof. For $\lambda_\sigma \to \infty$ and $\lambda_\alpha \bar\varepsilon = 0$, (31) reduces to

$$\underset{u_f, y_f^d, \alpha, \sigma}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix}\right) \tag{33a}$$

$$\text{s.t.} \quad \begin{bmatrix} z_{init} + \mathbb{1}_y \sigma_{init} \\ u_f \\ y_f^d + \sigma_y \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} \alpha, \tag{33b}$$

$$\begin{bmatrix} u(k) \\ y^d(k) \end{bmatrix} = \begin{bmatrix} u_r \\ y_r \end{bmatrix}, \quad k \in [t + T - \rho, t + T), \tag{33c}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t + T), \tag{33d}$$

$$\sigma = 0, \tag{33e}$$

where the constraint in (31e) can be replaced with (33e), independently from $\bar\varepsilon$. Let us now decompose the prediction model in (31b) as follows:

$$\begin{bmatrix} z_{init} \\ u_f \end{bmatrix} + \begin{bmatrix} \mathbb{1}_y \sigma_{init} \\ 0 \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \alpha, \tag{34}$$

$$y_f^d + \sigma_y = Y_F \alpha, \tag{35}$$

and Y_F as $Y_F := \hat{Y}_F + \bar{Y}_F$, with $\hat{Y}_F = \Pi_{Z_P, U_F}(Y_F)$ and $\bar{Y}_F = Y_F - \Pi_{Z_P, U_F}(Y_F)$. Leveraging on (33e), and the additional constraint

$$\sigma_y = Y_F(I - \Pi)\alpha = 0,$$

the relations in (34) and (35) can be rewritten as:

$$\alpha = \begin{bmatrix} Z_P \\ U_F \end{bmatrix}^\dagger \begin{bmatrix} z_{init} \\ u_f \end{bmatrix}, \tag{36}$$

$$y_f^d = Y_F \alpha = Y_F \Pi \alpha + Y_F(I - \Pi)\alpha = Y_F \Pi \alpha. \tag{37}$$

Note that (36) corresponds to (29b) in the constrained SPC problem, whereas (37) implies that the predictor $y_f^d = \hat{Y}_F \alpha$, thus concluding the proof. $\blacksquare$

The relationship established in Theorem 3 is consistent with the empirical evidences on the role of $\lambda_\alpha \bar\varepsilon$ and λ_σ discussed in Berberich et al. (2020, Section V). Although providing a guideline for the choice of these two hyperparameters, with the choice of λ_α inherently connected with the noise bound $\bar\varepsilon$, it is worth stressing that this choice will be optimal asymptotically, i.e., when ρ is selected according to the Akaike's criterion and $N_{data} \to \infty$.

4.3. DeePC with elastic net regularization

We now consider the problem with elastic net regularization in Dörfler et al. (2022, Section IV.D), that we rewrite for the control problem considered in this work by using our notation as follows:

$$\underset{u_f, y_f^d, \alpha}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix}\right) + \lambda_1 \|\alpha\|_1 + \lambda_2 \|(I - \Pi)\alpha\|_p \tag{38a}$$

$$\text{s.t.} \quad \begin{bmatrix} z_{init} \\ u_f \\ y_f^d \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} \alpha, \tag{38b}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t + T), \tag{38c}$$

where Π has been defined in (32). The following proposition provides the connection between the problem in (38) and the one in (29).

Theorem 4 (*SPC-based Regularization*). *Assuming the cost $J(\cdot)$ in (38a) is equal to (29e), then the solution to problem (29) coincides with the one of (38) for $\lambda_1 = 0$ and $\lambda_2 \to \infty$.*

Proof. For $\lambda_1 = 0$ and $\lambda_2 \to +\infty$, problem (38) reduces to

$$\underset{u_f, y_f^d, \alpha}{\text{minimize}} \quad J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix}\right) \tag{39a}$$

$$\text{s.t.} \quad \begin{bmatrix} z_{init} \\ u_f \\ y_f^d \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} \alpha, \quad \|(I - \Pi)\alpha\| = 0 \tag{39b}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t + T). \tag{39c}$$

In addition, by decomposing $Y_F := \hat{Y}_F + \tilde{Y}_F$, where $\hat{Y}_F = \Pi_{Z_P, U_F}(Y_F)$, and $\tilde{Y}_F = Y_F - \Pi_{Z_P, U_F}(Y_F)$, we have that

$$\hat{Y}_F = Y_F \Pi \quad \tilde{Y}_F = Y_F(I - \Pi).$$

Then, when $(I - \Pi)\alpha = 0$, we have

$$Y_F \alpha = Y_F \Pi \alpha + Y_F(I - \Pi)\alpha = Y_F \Pi \alpha = \hat{Y}_F \alpha.$$

$\blacksquare$

This result not only shows the connection between the control problem considered in this work and the regularized one proposed in Dörfler et al. (2022), but it also puts the results shown in Dörfler et al. (2022), where the role of λ_1 and λ_2 is evaluated experimentally, into a rigorous frame. Note that, since the performance of SPC is influenced by the choice of ρ in (11) and the dimension of the Hankel matrix (specifically the number of its columns N, see (21)), the choice of the regularization weights via Theorem 4 is likely to be optimal when ρ is chosen according to the Akaike's criterion and $N_{data} \to \infty$ (i.e., $N \to \infty$).

5. The γ-DDPC scheme

In this Section, we reformulate problem (29) by exploiting the LQ decomposition of the Hankel data matrices. On the one hand, this procedure leads to an even closer connection with subspace identification. On the other, it allows us to parametrize the solution to (29) in terms of a lower dimensional parameter vector. We thus consider the LQ decomposition of the joint input–output block Hankel matrix Z_{data} in (23), namely:

$$\begin{bmatrix} Z_P \\ U_F \\ Y_F \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}, \tag{40}$$

where the matrices $\{L_{ii}\}_{i=1}^3$ are all non-singular (under the assumptions of Lemma 3) and Q_i have orthonormal rows, i.e. *i.e.*, $Q_i Q_i^\top = I$, for $i = 1, \ldots, 3$, $Q_i Q_j^\top = 0$, $i \neq j$.

First of all, let us observe that $\hat{Y}_F := \Pi_{Z_P, U_F}(Y_F)$ in Lemma 2 can be expressed in terms of the LQ decomposition (40) as:

$$\hat{Y}_F = \begin{bmatrix} L_{31} & L_{32} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}. \tag{41}$$

By exploiting (40) and (41), we can thus express the constraint in (27) as follows:

$$z_{init} = Z_P \alpha = L_{11} Q_1 \alpha \tag{42a}$$

$$u_f = U_F \alpha = \begin{bmatrix} L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \alpha, \tag{42b}$$

where (42a) accounts for the initial condition of the predictive control problem, whereas (42b) links the optimal α with the control input. The predicted output in (26) can then be rewritten as

$$\hat{y}_f^d = \hat{Y}_F \alpha = \begin{bmatrix} L_{31} & L_{32} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \alpha^\star, \tag{43}$$

where $\alpha^\star$ is the minimum-norm solution to (42).

We can now leverage on triangular structure of (42) to characterize the minimum-norm solution $\alpha^\star$. In particular, (42a) always admits a solution (see Lemma 3 and Remark 3), that satisfies the following property.

Lemma 5 (*Definition of γ_1*). Let $\alpha_{init}^\star \in \mathbb{R}^N$ be the minimum-norm α solving (42a). Then, by defining $\gamma_1^\star \in \mathbb{R}^{(m+p)\rho}$ as the unique solution of

$$z_{init} = L_{11}\gamma_1, \tag{44}$$

$\alpha_{init}^\star$ can be written as $\alpha_{init}^\star = Q_1^\top \gamma_1^\star$, so that

$$\alpha_{init}^\star \in colspan\left(Q_1^\top\right).$$

Proof. Since Z_P has full column rank, so does L_{11} and any solution α to (42a) must satisfy

$$Q_1 \alpha = L_{11}^{-1} z_{init} = \gamma_1^\star.$$

The minimum-norm solution $\alpha_{init}^\star$ can be found by as

$$\alpha_{init}^\star = Q_1^\dagger \gamma_1^\star = Q_1^\top \gamma_1^\star,$$

thus concluding the proof.

Exploiting the definition of $\gamma_1^\star$ in Lemma 5, the control sequence u_f in (42b) can be equivalently written as

$$\begin{aligned} u_f &= L_{21}\gamma_1^\star + L_{22}\gamma_2 \\ \gamma_2 &= Q_2\alpha \end{aligned} \tag{45}$$

Based on this representation, we can provide additional insights on $\alpha^\star$, through the following result.

Lemma 6 (*Definition of γ_2*). Let $\alpha_f^\star \in \mathbb{R}^N$ indicate the minimum-norm α solving (45). Accordingly, define $\gamma_2^\star \in \mathbb{R}^{mT}$ as the unique solution of the least squares problem

$$L_{22}\gamma_2 = u_f - L_{21}\gamma_1^\star \tag{46}$$

where $\gamma_1^\star$ is defined in Lemma 5. Then $\alpha_f^\star$ can be written as:

$$\alpha_f^\star = Q_2^\top \gamma_2^\star, \tag{47}$$

so that

$$\alpha_f^\star \in colspan\left(Q_2^\top\right).$$

Proof. Since the matrix $[Z_P^\top \ U_F^\top]^\top$ has full rank, also L_{22} has full rank and is thus invertible. Any solution α to

$$u_f = L_{21}Q_1\gamma_1^\star + L_{22}Q_2\alpha$$

must therefore satisfy

$$Q_2\alpha = \underbrace{L_{22}^{-1}\left[u_f - L_{21}\gamma_1^\star\right]}_{:=\gamma_2^\star}.$$

and the minimum-norm solution is given by

$$\alpha_f^\star = Q_2^\dagger L_{22}^{-1}\left[u_f - L_{21}\gamma_1^\star\right] = Q_2^\top L_{22}^{-1}\left[u_f - L_{21}\gamma_1^\star\right] = Q_2^\top \gamma_2^\star.$$

From Lemmas 5 and 6, we can then characterize the minimum-norm parameter α of the whole behavioral model in (29b)–(29c) as follows.

Theorem 5 (*Decomposition of $\alpha^\star$*). Let $\alpha_{init}^\star \in \mathbb{R}^N$ and $\alpha_f^\star \in \mathbb{R}^N$ be defined as in Lemmas 5 and 6, respectively. Then, they satisfy the following properties:

1. $\alpha_{init}^\star = Q_1^\top \gamma_1^\star$;
2. $\alpha_f^\star = Q_2^\top \gamma_2^\star$;
3. $\alpha_{init}^\star$ is orthogonal to $\alpha_f^\star$;
4. $Q_1\alpha_f^\star = 0$ and $Q_2\alpha_{init}^\star = 0$
5. $Q_3\alpha_f^\star = Q_3\alpha_{init}^\star = 0$.

Therefore, $\alpha^\star = \alpha_{init}^\star + \alpha_f^\star$ is the minimum-norm vector satisfying the conditions:

$$\begin{aligned} \begin{bmatrix} z_{init} \\ u_f \end{bmatrix} &= \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \alpha^\star = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \alpha^\star \\ &= \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} \gamma_1^\star \\ \gamma_2^\star \end{bmatrix}, \end{aligned} \tag{48a}$$

$$\hat{y}_f^d = \sum_{i=1}^{3} L_{3i}Q_i\alpha^* = \sum_{i=1}^{2} L_{3i}Q_i\alpha^* = \sum_{i=1}^{2} L_{3i}\gamma_i^*. \tag{48b}$$

Proof. Conditions 1 and 2 have been proved in Lemmas 5 and 6 respectively. Condition 3 and 4 are direct consequences of the fact that $Q_1^\top Q_2 = 0$. Finally, Condition 5 derives from the fact that $Q_3^\top Q_i = 0$, for $i = 1, 2$. It is also straightforward to verify that, indeed, $\alpha^\star = \alpha_{init}^\star + \alpha_f^\star$ is a solution to (48). The fact that it is the minimum-norm solution derives from the fact that $\alpha^\star$ belongs to the column space of $[Q_1^\top \ Q_2^\top]$.

The properties highlighted above allow us to reformulate the DDPC problem as follows:

$$\underset{\gamma_1,\gamma_2}{\text{minimize}} \ J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix}\right) \tag{49a}$$

$$\text{s.t.} \ \begin{bmatrix} z_{init} \\ u_f \\ y_f^d \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \\ L_{31} & L_{32} \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix}, \tag{49b}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t+T), \tag{49c}$$

where the cost is defined in (29e) and we reshape the predictor based on the properties of the minimum-norm α highlighted in Theorem 5.

By looking at (49), it can be easily noticed that the cost and the value constraints in (49c) are independent of γ_1. In turn, γ_1 is solely determined by the initial conditions z_{init}. As such, γ_1 is not a proper optimization variable, but acts as a constraint that can be explicitly solved by setting:

$$\gamma_1^\star = L_{11}^{-1} z_{init}. \tag{50}$$

By leveraging LQ-decomposition, the problem of matching initial conditions can thus be decoupled from that of designing the optimal input. It is worth stressing once more that, according
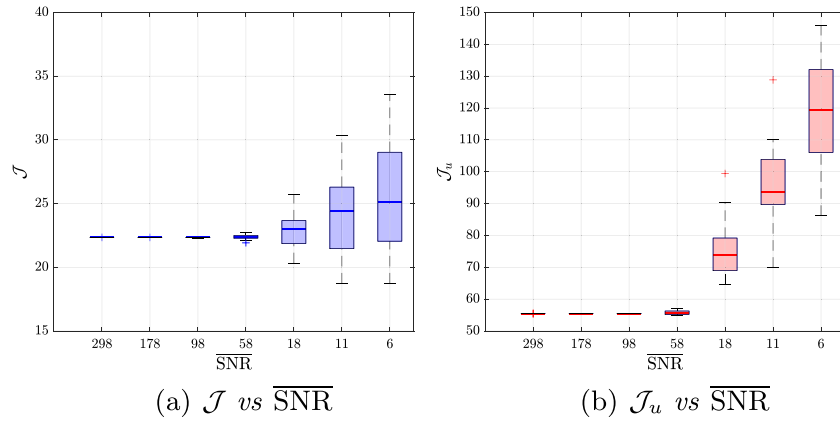
(a) $\mathcal{J}$ *vs* $\overline{\text{SNR}}$

(b) $\mathcal{J}_u$ *vs* $\overline{\text{SNR}}$

Fig. 1. Closed-loop validation tests: performance indexes *vs.* average *signal-to-noise ratio* ($\overline{\text{SNR}}$) over 30 Monte Carlo predictors.

Algorithm 1 γ-DDPC at time t

Input: Matrices $\{L_{i,j}\}_{i=1}^3$, $j = 1, 2$; penalties $Q \succeq 0$, $R \succ 0$; target y_r; constraint sets $\mathcal{U}$ and $\mathcal{Y}$; initial conditions z_{init}.

1. **Find** $\gamma_1^\star$ via (50);
2. **Optimize** γ_2 by solving (51);
3. **Construct** u_f according to (46);
4. **Extract** the first optimal input from u_f.

Output: Optimal input $u^\star(t)$.

---

to Lemma 5, $\gamma_1^\star$ and γ_2 found through (50) coincides with the ones leading to the minimum-norm α satisfying the initial conditions.

The constrained optimization problem to be solved at each time instant thus results into a reduced problem on γ_2 only, i.e.,

$$\underset{\gamma_2}{\text{minimize}} \ J\left(\begin{bmatrix} y_f^d \\ u_f \end{bmatrix}\right) \tag{51a}$$

$$\text{s.t.} \ \begin{bmatrix} u_f \\ y_f \end{bmatrix} = \begin{bmatrix} L_{21} & L_{22} \\ L_{31} & L_{32} \end{bmatrix} \begin{bmatrix} \gamma_1^\star \\ \gamma_2 \end{bmatrix}, \tag{51b}$$

$$u(k) \in \mathcal{U}, \ y^d(k) \in \mathcal{Y}, \ k \in [t, t + T), \tag{51c}$$

with $\gamma_1(t)$ fixed at the solution of (50).

According to this decomposition, we propose the γ-DDPC scheme, summarized in Algorithm 1. Apart from inheriting the properties of the predictor highlighted using the LQ decomposition with respect to noise handling, the γ-DDPC scheme is likely to be computationally advantageous. Indeed, the dimension of the optimization variable $\gamma_2 \in \mathbb{R}^{mT}$ in (51) is likely to be considerably smaller than the one of $\alpha \in \mathbb{R}^N$. At the same time, retrieving γ_1 requires the inversion of a matrix with dimensions dictated by the chosen ρ.

Remark 4 (*Choice of ρ (Part II)*). The length of the "past" window plays a pivotal role in shaping the performance of the predictive controller. On the one hand, ρ should be chosen by following an identification-oriented reasoning (see Remark 2). On the other, a smaller ρ reduces the number of data needed to solve the DDPC problem ($N_{data} := N + T + \rho$), and it would result in a computationally lighter DDPC problem. Its value has thus to be selected by trading-off between these requirements.

5.1. Explaining regularization in DDPC

By looking at the DDPC problem from a different angle, the results presented so far allow us to have a clearer vision on the

actual effect that additional regularization terms have on the optimal control action generated when solving (29). We stress that the use of regularization is currently, by and large, the strategy proposed by most of the literature to cope with stochastic noise in DDPC.

The properties highlighted in Theorem 5 indicate that $Q_3\alpha$ should be set to zero, if one seeks to reduce the effect of noise on the predictions exploited to determine the optimal control action. At the same time, one should not excessively shrink the values of $Q_i\alpha$, for $i = 1, 2$. While these two conflicting requirements on α can be easily accommodated when decomposing the predictor using the LQ decomposition, this operation is not as easy when the predictor in (29b)–(29c) is used as it is. Indeed, in this last case, one can only try shrink the whole vector α by introducing a regularizer in the cost, as already proposed in Berberich et al. (2020) and Dörfler et al. (2022). Although such procedure has proven to be effective, the regularization strength has to be well calibrated to trade-off between reducing the norm of α and retaining the information needed to produce a meaningful control action. In turn, achieving this balance requires the fine tuning of the regularization penalty, representing a well-known drawback of regularization-based DDPC approaches. Indeed, existing procedures generally require closed-loop experiments to calibrate the regularization parameters, which can endanger the safety of the plant, ultimately limiting the applicability of existing DDPC strategies.

6. A benchmark case study

To assess the effectiveness of the proposed γ-DDPC scheme, while validating the conclusions drawn in Section 5.1, we consider the same benchmark example proposed in Bemporad, Morari, Dua, and Pistikopoulos (2002). Therefore, the unknown plant to be controlled is described by the following model:

$$\begin{cases} x(t + 1) = \begin{bmatrix} 0.7326 & -0.0861 \\ 0.1722 & 0.9909 \end{bmatrix} x(t) + \begin{bmatrix} 0.0609 \\ 0.0064 \end{bmatrix} u(t) + Ke(t), \\ y(t) = \begin{bmatrix} 0 & 1.4142 \end{bmatrix} x(t) + e(t), \end{cases} \tag{52}$$

where the innovation is set to be zero-mean and Gaussian distributed, while $K \in \mathbb{R}^{1 \times 2}$ is randomly chosen according to a normal distribution, with all the eigenvalues of $A - KC$ being inside the unit circle. By considering a prediction horizon of length $T = 40$ and $\rho = 23$ (selected according to Remark 2), we design the predictive controllers solving a zero regulation
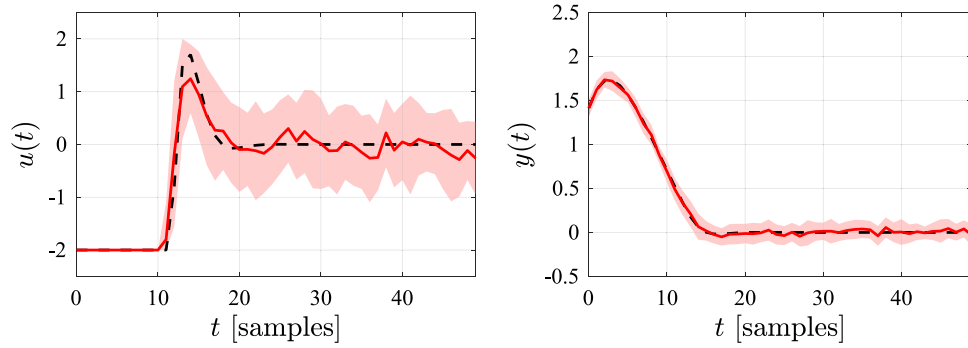
Fig. 2. Closed-loop validation tests ($\overline{\mathrm{SNR}}$ = 18 dB): average (red) closed-loop response and inputs with their standard deviations (shaded area) over the 30 predictors *vs.* oracle *noise-free* MPC (black dashed lines). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
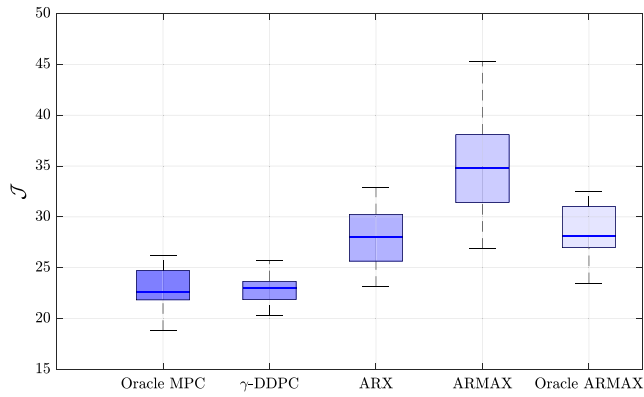

(a) $\mathcal{J}$ *vs* predictive strategy
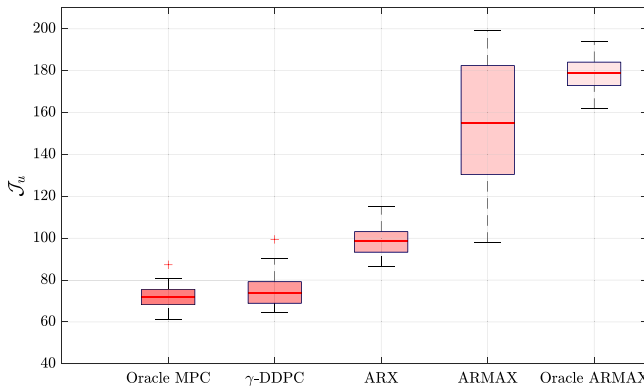
(b) $\mathcal{J}_u$ *vs* predictive strategy

Fig. 3. Closed-loop validation tests ($\overline{\mathrm{SNR}}$ = 18 dB): performance indexes *vs.* predictive strategy over 30 Monte Carlo predictors.

problem by running Algorithm 1[6] with $Q = I$, $R = 10^{-3}$ and $y_r = u_r = 0$, as in Bemporad et al. (2002). To have a quantitative assessment of performance, for all closed-loop tests we consider the following indexes:

$$\mathcal{J} = \sum_{t=0}^{T_v-1} \|y(t)\|_Q^2 + \|u(t)\|_R^2, \quad \mathcal{J}_u = \sum_{t=0}^{T_v-1} u(t)^2,$$

---

[6] All tests have been carried out on an M1 chip, running MATLAB 2021a, while the optimization problems are solved with CVX (Grant & Boyd, 2008, 2014).
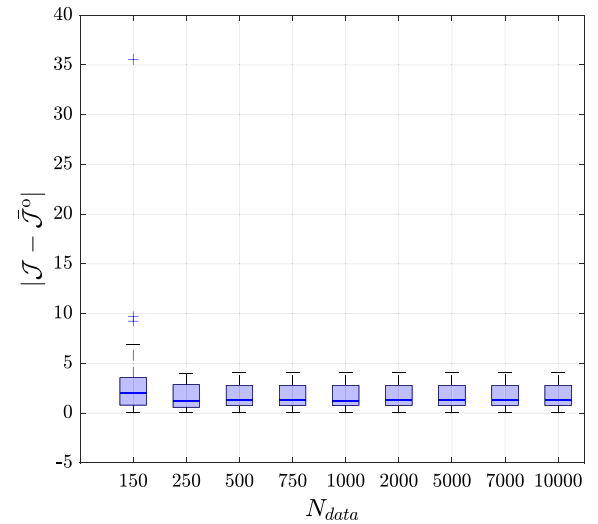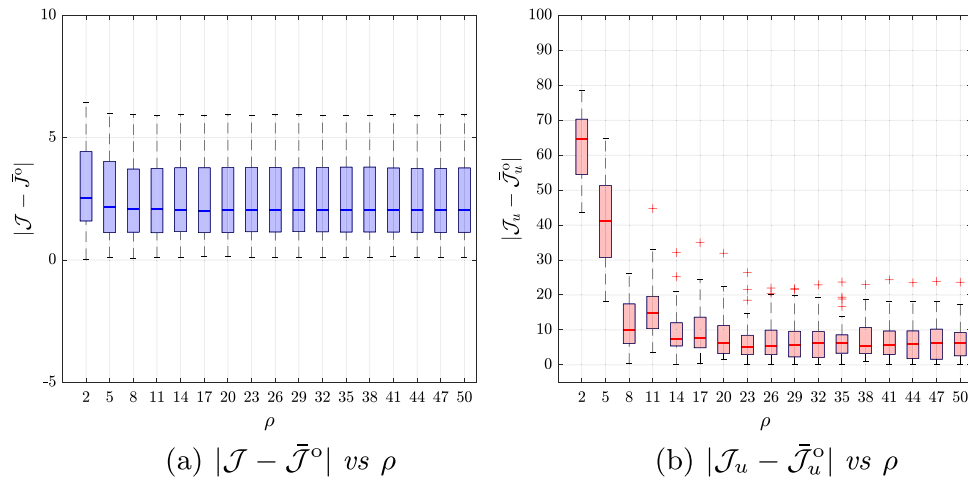

(a) $|\mathcal{J} - \bar{\mathcal{J}}^{\mathrm{o}}|$ *vs* N_{data}
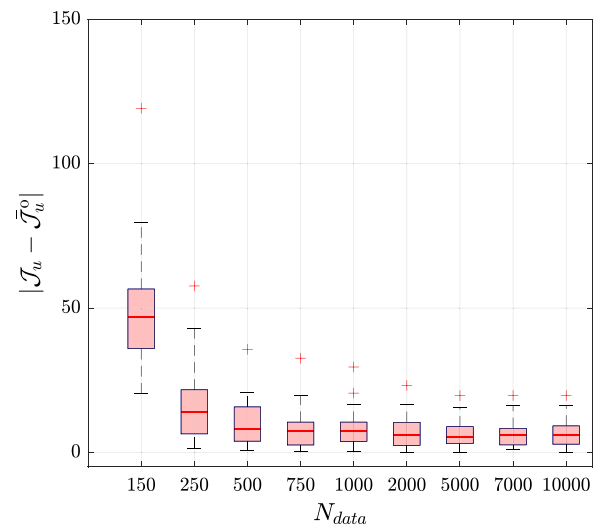
(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^{\mathrm{o}}|$ *vs* N_{data}

Fig. 4. Closed-loop validation tests ($\overline{\mathrm{SNR}}$ = 18 dB): absolute differences between the performance indexes of γ-DDPC and the average values of those associated with the *noisy oracle* MPC *vs.* N_{data} over 30 Monte Carlo predictors.

Fig. 5. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the performance indexes of γ-DDPC and the average values of those associated with the *noisy* oracle MPC *vs.* length of the "past horizon" ρ over 30 Monte Carlo predictors.
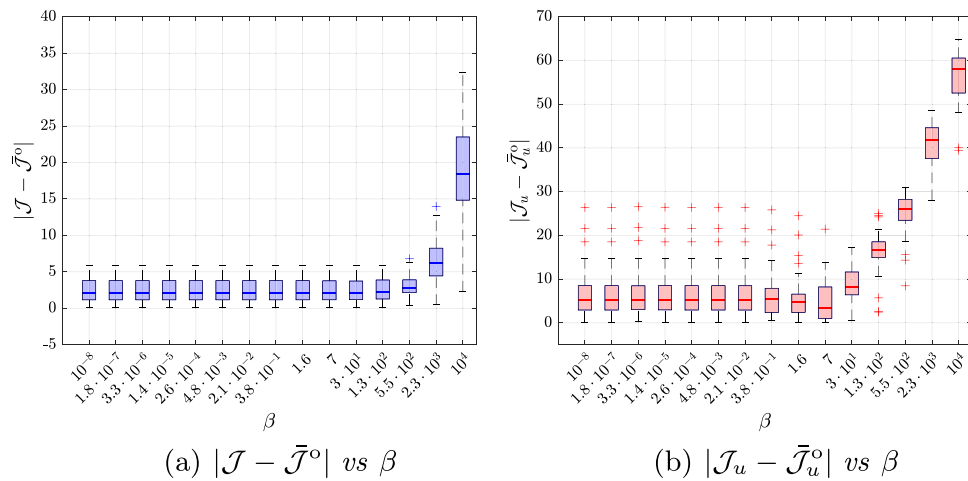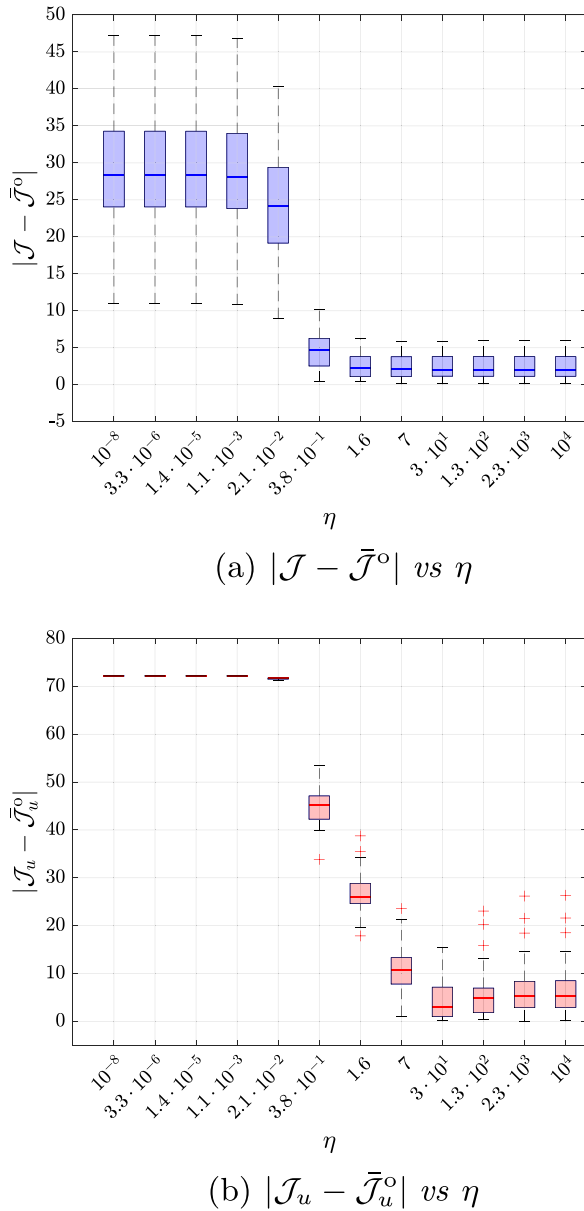

Fig. 6. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the performance indexes of γ-DDPC and the average values of those associated with the *noisy* oracle MPC *vs.* penalties β on a 2-norm regularization on γ_2 over 30 Monte Carlo predictors.

that allow us to have a compact information on the tracking performance and the input effort in testing.

We initially focus on assessing the performance of the γ-DDPC scheme introduced in Section 5. Firstly, we assess the sensitivity of γ-DDPC to noise in the available batch of data $\mathcal{D}_{N_{data}}$. By progressively increasing the level of noise, we thus perform 30 Monte Carlo simulations of length $N_{data} = 1000$ with a random input sequence, uniformly distributed in the interval $[-5, 5]$, to generate different datasets. Closed-loop performance is then evaluated for each predictive model and level of noise by using γ-DDPC to close the control loop over tests of length $T_v = 50$, always starting from the same initial condition. As shown in Fig. 1, both the performance and control effort are quite consistent when the average *signal-to-noise ratio* (SNR) is high. Instead, a slight degradation in performance is experienced when the average noise corrupting the data used to construct the predictor decreases, along with an increase in the control effort required during closed-loop testing. These results generally show that the proposed γ-DDPC strategy allows the closed-loop system to track (on average) the reference, in spite of the process and measurement noise affecting it. This consideration is further confirmed by the results reported in Fig. 2, where the closed-loop inputs and output attained with γ-DDPC are compared with the ones obtained via an MPC designed with the true system matrices (denominated from now on *oracle MPC*) within a noise-free setting.

For a fixed level of noise, we then compare the closed-loop performance achieved with Algorithm 1 with the ones attained by designing an MPC with an identified model[7] of the plant. To this end, we keep the input/output structure of the predictor by identifying both an *autoregressive* model *with exogenous inputs* (ARX) of order 23, an "oracle" *autoregressive moving average* model *with exogenous inputs* (ARMAX) of order 2 and ARMAX models with orders selected according to Remark 2.[8] As shown in Fig. 3, the use of all identified models tends to slightly deteriorate performance, while requiring an additional control effort. The main deterioration in performance is visible when the ARMAX models are used to design the MPC. These results are example-dependent and should not be generalized. However, we could indeed conclude that in some cases it *might* be preferable to use γ-DDPC over a identification+model-based control procedure. We also evaluate how γ-DDPC performs when increasing N_{data} over noisy closed-loop tests. As shown in Fig. 4, the difference between the overall cost and the required control effort tends to decrease with the number of data, in line with established results in system identification. Lastly, we assess the sensitivity

---

7 The model is identified via the prediction error method (Ljung, 1997).

8 The average order of the ARMAX models is 7, while its standard deviation is 5.

(a) $|\mathcal{J} - \bar{\mathcal{J}}^{\circ}|$ *vs* η

(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^{\circ}|$ *vs* η

Fig. 7. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the performance indexes of γ-DDPC and the average values of those associated with the *noisy* oracle MPC *vs.* penalties η on a 2-norm regularization on γ_3 over 30 Monte Carlo predictors.

of γ-DDPC to the only free parameter of this scheme, namely ρ. As shown in Fig. 5, the main changes due to different choices of the "past horizon" are visible in the index assessing the control effort. In particular, excessively small values of ρ results into the demand for a greater control effort than that required by the oracle MPC. By increasing ρ, the input effort required by γ-DDPC tends to become aligned with that associated with the oracle MPC, while slightly increasing again when $\rho > 30$. Note that $\mathcal{J}_u$ gets the closest to the average input effort index of the oracle MPC for $\rho = 23$, thus validating the choice we have automatically performed through the Akaike's criterion.

6.1. Effect of additional regularization on γ-DDPC

By keeping the level of noise acting on the batch and online data, we now study the effect of an additional 2-norm regularization on γ_2, with $\beta > 0$ indicating the associated penalty. As

shown in Fig. 6, the performance index $\mathcal{J}$ tends to be rather insensitive to the additional regularization term up to a certain value of β. However, when β increases sufficiently, performance tends to deteriorate, while the input effort tends to consistently increase with respect to the oracle MPC one. Since such a behavior is certainly undesirable, this result validates in this experimental case the claims in Section 5.1. Indeed, the additional regularization leads to a deterioration of performance, likely to be induced by the change that the regularization enforces on the actual performance-oriented cost. To prove the effectiveness of our structural choices, within the same framework we consider the DDPC problem with the predicted output defined as

$$y_f = \sum_{i=1}^{3} L_{3i}\gamma_i,$$
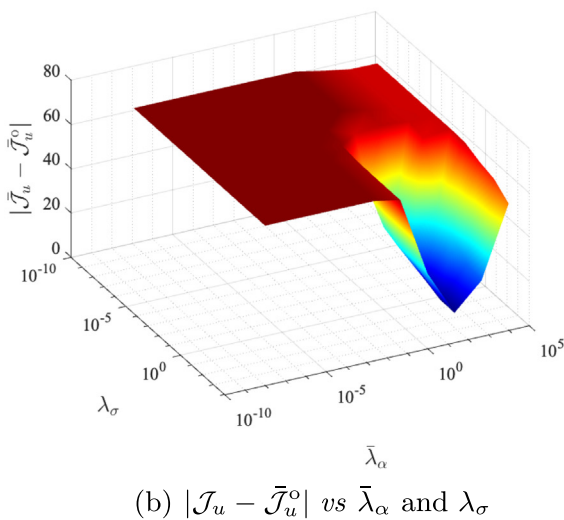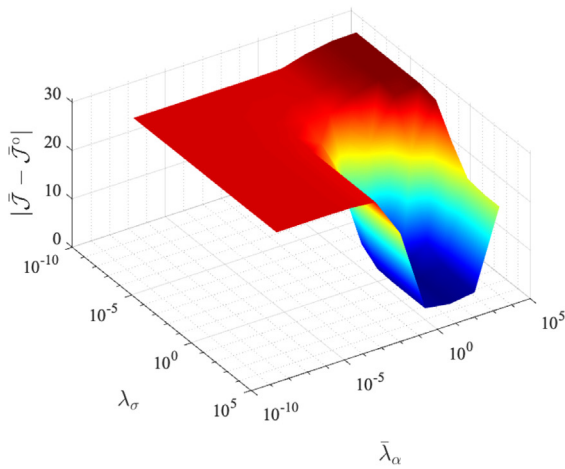
and γ_3 not set to zero beforehand, as in the proposed γ-DDPC approach. In this case, γ_3 is steered towards small values via an addition of a 2-norm regularization term in the cost weighted by $\eta > 0$. As shown in Fig. 7, only by heavily regularizing γ_3 we obtain performance comparable with the ones obtained with the oracle predictive controller. Specifically low η result in an ineffective DDPC scheme, with the system actually operating in open-loop. These results once again show the expected detrimental effect of poor choices of the regularization parameter, highlighting the advantages of embedding insights given by subspace identification into the predictor used in the DDPC scheme.

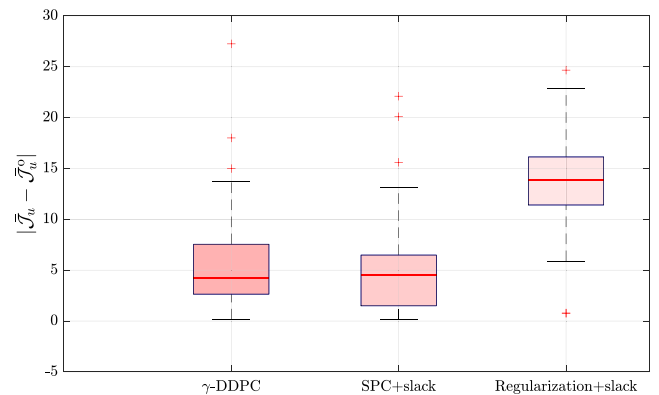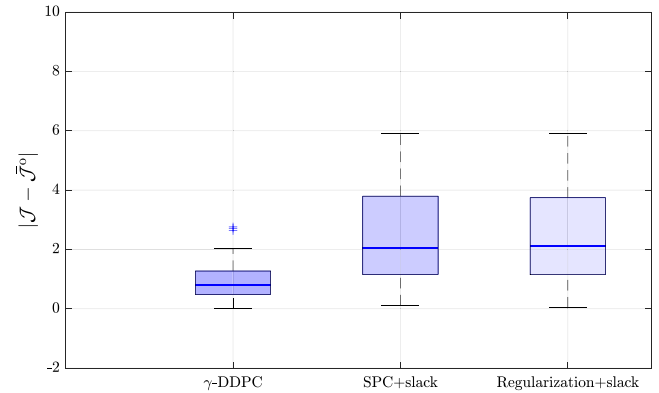### 6.2. Validating results on regularized DDPC schemes

We now analyze the sensitivity of the three regularized DDPC approaches considered in Section 4 to different choices of their main tuning knobs, with the aim of experimentally validating the results stemming from the derived unified framework. The behavior of the performance indexes shown in Fig. 8 supports our conclusions. Indeed, the SPC+slack scheme proposed in Fiedler and Lucia (2021) tends to behave more closely to the oracle MPC for growing λ. Meanwhile, the input sequence fed to the system in closed loop tends to become equal to zero when λ is small, concurrently causing a deterioration of the overall closed-loop performance. As shown in Fig. 9, the choice of the regularization parameters is crucial to attain satisfactory performance when exploiting the approach proposed in Berberich et al. (2020), balancing the need to have a meaningful control action and the one of rejecting noise. The attained behavior validates the conclusions drawn in Section 4 with respect to the penalty λ_σ in (31). Indeed, higher values of this weight tends to improve the overall performance of the closed-loop. At the same time, since the dataset is finite and noisy, the results in Fig. 9 highlight the importance of regularization for this DDPC formulation. Moreover, these results show that regularizing the whole parameter vector α, along with introducing a set of slacks, requires a careful selection of both the associated the regularization penalty. When compared with γ-DDPC, even with the best possible tuning, the schemes presented in Berberich et al. (2020) and Fiedler and Lucia (2021) result in the worst average performance with respect to the oracle MPC and a higher variability of the closed-loop behavior, see Fig. 10(a). Note that, when the regularization penalty is properly tuned, the introduction of the slack variables in (30) leads to an overall control effort similar to the oracle input sequence. Lastly, Figs. 11 and 12 corroborate the conclusions drawn in Section 4. Indeed, it is clear that larger values of λ_2 and smaller λ_1 lead to performance that are comparable with that of the γ-DDPC. In particular, for $\lambda_1 = 10^{-8}$, it is clear that λ_2 has an effect similar to the one of η and that (as expected) it is advisable to set λ_2 as large as possible.

(a) $|\mathcal{J} - \bar{\mathcal{J}}^\circ|$ *vs* λ

(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^\circ|$ *vs* λ
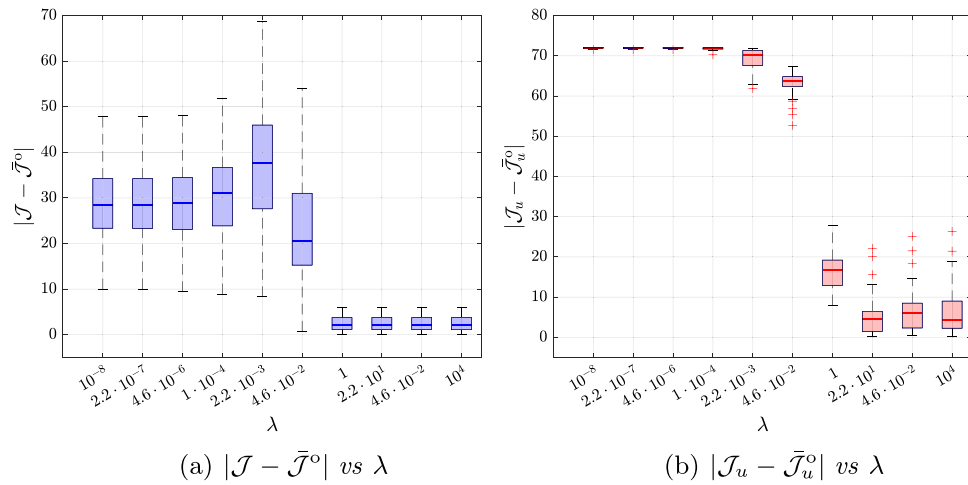
Fig. 8. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the performance indexes attained with Fiedler and Lucia (2021) and the average values of those associated with the *noisy* oracle MPC *vs.* penalties λ over 30 Monte Carlo predictors.

(a) $|\mathcal{J} - \bar{\mathcal{J}}^\circ|$ *vs* $\bar{\lambda}_\alpha$ and λ_σ

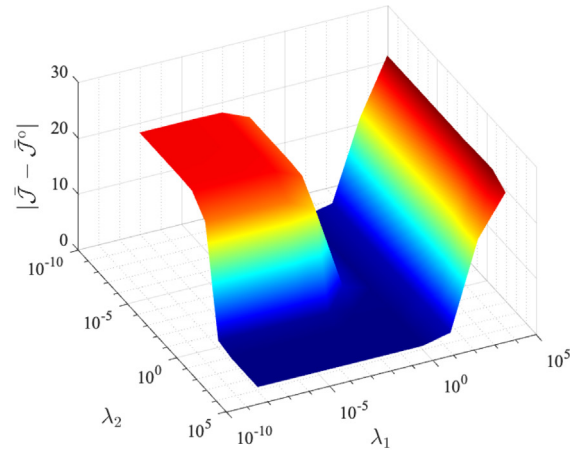(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^\circ|$ *vs* $\bar{\lambda}_\alpha$ and λ_σ

Fig. 9. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the performance indexes attained with Berberich et al. (2020) and the average values of those associated with the *noisy* oracle MPC *vs.* penalties $\bar{\lambda}_\alpha = \lambda_\alpha \bar{\varepsilon}$ and λ_σ over 30 Monte Carlo predictors.

(a) $|\mathcal{J} - \bar{\mathcal{J}}^\circ|$ *vs* approaches

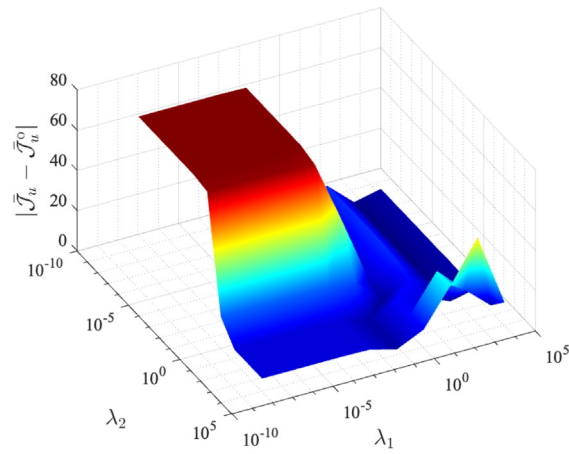(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^\circ|$ *vs* approaches

Fig. 10. Closed-loop validation tests ($\overline{\text{SNR}} = 18$ dB): absolute differences between the average performance indexes attained with he *noisy* oracle MPC and $\mathcal{J}$ and $\mathcal{J}_u$ obtained with γ-DDPC, and the SPC+slack scheme in (30) and the regularized approach with slack in (31) with the best tuning over 30 Monte Carlo predictors..

7. Conclusions

In this paper, exploiting subspace identification tools, we have provided a unifying framework for several regularized data-driven predictive control schemes proposed in the literature,
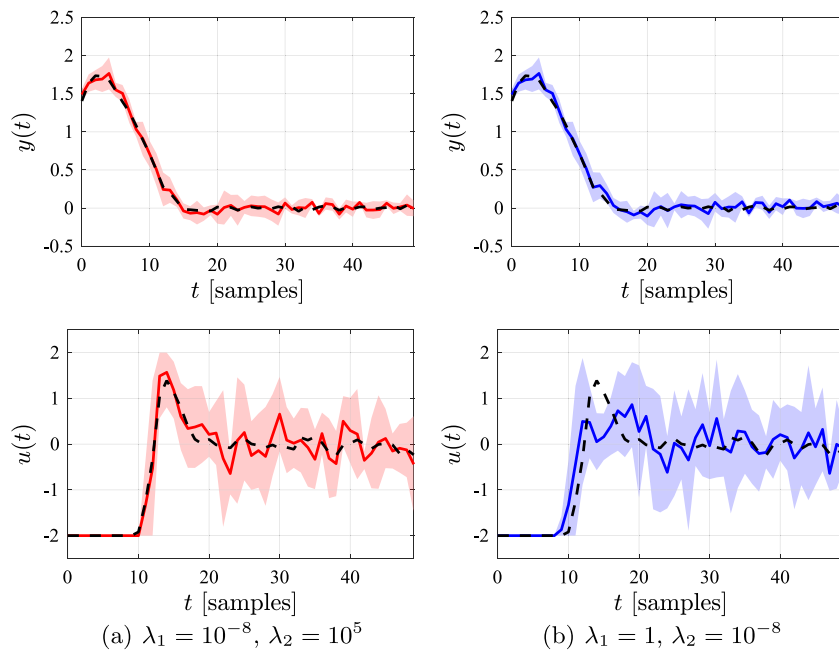
(a) $|\mathcal{J} - \bar{\mathcal{J}}^\circ|$ *vs* λ_1 and λ_2

(b) $|\mathcal{J}_u - \bar{\mathcal{J}}_u^\circ|$ *vs* λ_1 and λ_2

Fig. 11. Closed-loop validation tests ($\overline{\mathrm{SNR}} = 18$ dB): absolute differences between the performance indexes attained with Dörfler et al. (2022) and the average values of those associated with the *noisy* oracle MPC *vs.* penalties λ_1 and λ_2 over 5 Monte Carlo predictors.

(a) $\lambda_1 = 10^{-8}$, $\lambda_2 = 10^5$

(b) $\lambda_1 = 1$, $\lambda_2 = 10^{-8}$

Fig. 12. Average inputs for closed-loop validation tests ($\overline{\mathrm{SNR}} = 18$ dB): oracle (dashed black line) *vs.* average input (line) and standard deviation (shaded area) for different values of λ_1 and λ_2 in (38) over 5 Monte Carlo predictors.

showing that they can be seen as variations of subspace predictive control. This result extends the validity of these approaches beyond scenarios in which only measurement noise affects the system under control. As a by-product, we have discussed the role of regularization, which is generally advocated in the literature as a tool to extend deterministic ideas to the noisy setting. By relying on the predictor decomposition proposed in the paper, we have further introduced the γ-DDPC problem, leading to a two-stage scheme where the effect of initial conditions, performance objectives and constraints is accounted for by solving two smaller optimization problems. By means of a numerical example, we show how the formulation at the core of γ-DDPC eases the interpretation of the effect of different regularization terms on the closed-loop behavior of the system, while validating the outcome of our discussions about the selection of the regularization penalties.

Future works will be devoted to the analysis of the closed loop properties of γ-DDPC, and to extend the latter to explicitly account for the error induced by the availability of a finite dataset. In addition, we will analyze the impact of regularization when regularized DDPC schemes are applied to nonlinear systems.

References

Akaike, H. (1969). Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, 21, 243–247.

Alanwar, Amr, Stürz, Yvonne, & Johansson, Karl Henrik (2021). Robust data-driven predictive control using reachability analysis. arXiv preprint arXiv:2103.14110.

Alpago, Daniele, Dörfler, Florian, & Lygeros, John (2020). An extended Kalman filter for data-enabled predictive control. *IEEE Control Systems Letters*, 4(4), 994–999.

Bauer, D. (2005). Asymptotic properties of subspace estimators. *Automatica*, 41, 359–376.

Bauer, D., & Jansson, M. (2000). Analysis of the asymptotic properties of the MOESP type of subspace algorithms. *Automatica*, 36, 497–509.

Bemporad, Alberto, Morari, Manfred, Dua, Vivek, & Pistikopoulos, Efstratios N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1), 3–20.

Berberich, Julian, Köhler, Johannes, Müller, Matthias A., & Allgöwer, Frank (2020). Data-driven model predictive control with stability and robustness guarantees. *IEEE Transactions on Automatic Control*, 66(4), 1702–1717.

Breschi, Valentina, & Formentin, Simone (2020). Direct data-driven design of switching controllers. *International Journal of Robust and Nonlinear Control*, 30(15), 6042–6072.

Chiuso, A. (2006). Asymptotic variance of closed-loop subspace identification algorithms. *IEEE Transactions on Automatic Control*, 51(8), 1299–1314.

Chiuso, A. (2007a). On the relation between CCA and predictor-based subspace identification. *IEEE Transactions on Automatic Control*, 52(10), 1795–1812.

Chiuso, A. (2007b). The role of vector autoregressive modeling in predictor based subspace identification. *Automatica*, 43(6), 1034–1048.

Chiuso, Alessandro (2010). On the asymptotic properties of closed-loop CCA-type subspace algorithms: Equivalence results and role of the future horizon. *IEEE Transactions on Automatic Control*, 55(3), 634–649.

Chiuso, A., & Picci, G. (2005). Consistency analysis of some closed-loop subspace identification methods. *Automatica*, 41(3), 377–391.

Coulson, Jeremy, Lygeros, John, & Dörfler, Florian (2019). Data-enabled predictive control: In the shallows of the DeePC. In *2019 18th European control conference* ECC, (pp. 307–312). IEEE.

Coulson, Jeremy, Lygeros, John, & Dörfler, Florian (2019). Regularized and distributionally robust data-enabled predictive control. In *2019 IEEE 58th conference on decision and control* CDC, (pp. 2696–2701).

Dahlén, A., & Scherrer, W. (2004). The relation of CCA subspace method to a balanced reduction of an autoregressive model. *Journal of Econometrics*, 118(1–2), 293–312.

Dörfler, Florian, Coulson, Jeremy, & Markovsky, Ivan (2022). Bridging direct & indirect data-driven control formulations via regularizations and relaxations. *IEEE Transactions on Automatic Control*.

Favoreel, Wouter, Moor, Bart De, & Gevers, Michel (1999). SPC: Subspace predictive control. *IFAC Proceedings Volumes*, 32(2), 4004–4009, 14th IFAC World Congress 1999, Beijing, Chia, 5-9 July.

Fiedler, Felix, & Lucia, Sergio (2021). On the relationship between data-enabled predictive control and subspace predictive control. In *2021 European control conference* ECC, (pp. 222–229).

Formentin, Simone, Campi, Marco C., Carè, Algo, & Savaresi, Sergio M. (2019). Deterministic continuous-time Virtual Reference Feedback Tuning (VRFT) with application to PID design. *Systems & Control Letters*, 127, 25–34.

Formentin, Simone, & Karimi, Alireza (2012). A data-driven approach to mixed-sensitivity control with application to an active suspension system. *IEEE Transactions on Industrial Informatics*, 9(4), 2293–2300.

Formentin, Simone, Van Heusden, Klaske, & Karimi, Alireza (2014). A comparison of model-based and data-driven controller tuning. *International Journal of Adaptive Control and Signal Processing*, 28(10), 882–897.

Gevers, Michel (2005). Identification for control: From the early achievements to the revival of experiment design. *European Journal of Control*, 11(4–5), 335–352.

Grant, Michael, & Boyd, Stephen (2008). Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, & H. Kimura (Eds.), *Lecture notes in control and information sciences*, *Recent advances in learning and control* (pp. 95–110). Springer-Verlag Limited.

Grant, Michael, & Boyd, Stephen (2014). CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx.

Hannan, E. J., & Poskitt, D. S. (1988). Unit canonical correlations between future and past. *The Annals of Statistics*, 16, 784–790.

Hjalmarsson, H. (2005). From experiment design to closed-loop control. *Automatica*, 41(3), 393–438.

Karimi, Alireza, & Kammer, Christoph (2017). A data-driven approach to robust control of multivariable systems by convex optimization. *Automatica*, 85, 227–233.

Krishnan, Vishaal, & Pasqualetti, Fabio (2021). On direct vs indirect data-driven predictive control. arXiv preprint arXiv:2103.14936.

Ljung, L. (1997). *System identification, theory for the user*. Prentice Hall.

Moonen, M., De Moor, B., Vandeberghe, L., & Vandewalle, J. (1989). On- and off-line identification of linear state-space models. *International Journal of Control*, 49(1), 219–232.

van der Vaart, A. W. (1998). *Asymptotic statistics*. Cambridge University Press.

Van Overschee, P., & De Moor, B. (1994). N4SID: Subspace algorithms for the identification of combined deterministic– stochastic systems. *Automatica*, 30, 75–93.

Willems, Jan C., Rapisarda, Paolo, Markovsky, Ivan, & De Moor, Bart L. M. (2005). A note on persistency of excitation. *Systems & Control Letters*, 54(4), 325–329.

Yin, Mingzhou, Iannelli, Andrea, & Smith, Roy S. (2021). Maximum likelihood estimation in data-driven modeling and control. *IEEE Transactions on Automatic Control*.

Ziegler, John G., Nichols, Nathaniel B., et al. (1942). Optimum settings for automatic controllers. *Transactions of the ASME*, 64(11).

Valentina Breschi received her Master's degree in Electrical and System Engineering in 2014 from the University of Florence, Italy. She received her Ph.D. in Control Engineering in 2018 from IMT School for Advanced Studies of Lucca, Italy. In 2017, she was a visiting scholar at the Department of Aerospace Engineering, University of Michigan, Ann Arbor. In 2018–2020, she held a postdoctoral position at Politecnico di Milano, where she is currently an Assistant Professor. Her main research interests include system identification, data-driven control, data-analysis and policy design for mobility systems.

Alessandro Chiuso is Professor with the Department of Information Engineering, Università di Padova. He received the "Laurea" degree summa cum laude in Telecommunication Engineering from the University of Padova in July 1996 and the Ph.D. degree (Dottorato di ricerca) in System Engineering from the University of Bologna in 2000. He has been a visiting scholar with the Dept. of Electrical Engineering, Washington University St. Louis and Post-Doctoral fellow with the Dept. Mathematics, Royal Institute of Technology, Sweden. He joined the University of Padova as an Assistant Professor in 2001, Associate Professor in 2006 and then Full Professor since 2017. He currently serves as Editor (System Identification and Filtering) for Automatica. He has served as an Associate Editor for Automatica, IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology, European Journal of Control and MCSS. He is chair of the IFAC Technical Committee on Modeling, Identification and Signal Processing. He has been General Chair of the IFAC Symposium on System Identification, 2021 and he is a Fellow of IEEE (Class 2022). His research interests are mainly at the intersection of Machine Learning, Estimation, Identification and their Applications, Computer Vision and Networked Estimation and Control.

Simone Formentin was born in Legnano, Italy, in 1984. He received his B.Sc. and M.Sc. degrees cum laude in Automation and Control Engineering from Politecnico di Milano, Italy, in 2006 and 2008, respectively. In 2012, he obtained his Ph.D. degree cum laude in Information Technology within a joint program between Politecnico di Milano and Johannes Kepler University of Linz, Austria. After that, he held two postdoctoral appointments at the Swiss Federal Institute of Technology of Lausanne (EPFL), Switzerland and the University of Bergamo, Italy, respectively. Since 2014, he has been with Politecnico di Milano, first as an assistant professor, then as an associate professor. He is the chair of the IEEE TC on System Identification and Adaptive Control, the social media representative of the IFAC TC on Robust Control and a member of the IFAC TC on Modelling, Identification and Signal Processing. He is an Associate Editor of Automatica and the European Journal of Control. His research interests include system identification and data-driven control with a focus on automotive and financial applications.