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

Valentina Breschi \textsuperscript{a}, Alessandro Chiuso \textsuperscript{b}, Simone Formentin \textsuperscript{a}

\textsuperscript{a} Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, P.zza L. Da Vinci, 32, 20133 Milano, Italy.

\textsuperscript{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 $\gamma$-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 $\gamma$-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.

Key words: data-based control, control of constrained systems, regularization, identification for control

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 [22]. 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 [23], and accurate modeling for control requires significant time and several (costly) technical expertise, 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 [27]. Early attempts in the direction of DDC date back to 1942, with the first studies by Ziegler and Nichols about PID auto-tuning [36]. 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., [20,8,21,28]. 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” [34] (or variations of the latter), most of existing DDPC techniques replace model equations with suitable data-based constraints\textsuperscript{1} (see e.g., [7,14]).

\textsuperscript{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
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 [29].

The transition 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 [7], 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 [2], 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 [35], 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 [14] and [7] is given in the recent paper [17]. 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 [15]. One of the regularized schemes proposed in [17] is then connected with Subspace Predictive Control (SPC) [18] by [19], where the introduction of additional slacks is further propose to cope with noisy online data. Kalman filter approaches have finally been suggested in [3] 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 [19,7,17] can all be recast into a unified framework, stemming from the constrained counterpart of the SPC scheme originally proposed in [18].

C2. Based on this unified framework, we discuss how the choice of key hyperparameters in [19,7,17] 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 $\gamma$-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 $\gamma$-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 details 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 $\gamma$-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 $\gamma$-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 $A^\top$.  

will simply talk about data-driven predictive control from now on.
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 $t > M$,

$$\|a(t)\| \leq c\|b(t)\|. \quad (1)$$

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 $t > M$,

$$\|a(t)\| \leq \epsilon\|b(t)\|. \quad (2)$$

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., [31]. Given $a$ and $b$, we use the symbols $\approx$ and $\equiv$ to denote equality up to $o_P(1/\sqrt{N})$ and up to $O_P(1/\sqrt{N})$, respectively. Namely

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

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

$\Pi_A[B]$ 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:

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

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

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

2 Setting and goal

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

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

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:

$$\begin{align*}
\min_{u(k),k \in [t,t+T]} & \frac{1}{2} \sum_{k=t}^{t+T-1} \mathbb{E} \left[ \|y(k) - y_r\|_Q^2 \right] + \|u(k) - u_r\|_R^2 \\
\text{s.t.} & \ x(k+1) = Ax(k) + Bu(k) + Ke(k), \ k \in [t,t+T], \\
& \ y(k) = Cx(k) + Du(k) + e(k), \ k \in [t,t+T], \\
& \ x(t) = x_{\text{init}}, \\
& \ u(k) \in U, \ \mathbb{E}[y(k)] \in Y, \ k \in [t,t+T],
\end{align*} \quad (7)$$

where $k \in \mathbb{Z}$, $x_{\text{init}}$ is the state at time $t$, $e(k)$ is a zero mean noise with variance $\text{Var}\{e(k)\}$, the sets $U$, $Y$ denote inputs and output constraints, and the expectation $\mathbb{E}[]$ is taken w.r.t. the future noise sequence $e(k)$, $k \in [t,t+T)$, and conditionally on the initial state $x_{\text{init}}$ 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 $D_{N,\text{data}} = \{u(j),y(j)\}_{j=1}^{N_{\text{data}}}$ collected in open loop 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 $e(k)$.

Extension to data collected in closed-loop is possible. Yet, for the sake of exposition, its treatment is deferred to future publications.
The initial state $x_{\text{init}}$ 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.,

$$
\mathbb{E} \left[ \| y(k) - y_r \|_2^2 \right] = \mathbb{E} \left[ \| y(k) - y_r \|_Q^2 \right] + \mathbb{E} \left[ \| y(k) - \mathbb{E} [ y(k) ] \|_2^2 \right]
$$

Since the variance term $\mathbb{E} \left[ \| y(k) - \mathbb{E} [ y(k) ] \|_2^2 \right]$ is independent of the input signal $u(k)$, only the conditional (given $x_{\text{init}}$ and $u(t)$) 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

$$
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \left[ \sum_{k=t}^{t+T-1} \| y^d(k) - y_r \|_Q^2 + \| u(k) - u_r \|_T^2 \right] \\
\text{subject to} & \quad x^d(k+1) = Ax^d(k) + Bu(k), \quad k \in [t, t+T), \quad (8a) \\
& \quad y^d(k) = Cx^d(k) + Du(k), \quad k \in [t, t+T), \quad (8b) \\
& \quad x^d(t) = x_{\text{init}}, \quad (8c) \\
& \quad u(k) \in \mathcal{U}, \quad y^d(k) \in \mathcal{Y}, \quad k \in [t, t+T). \quad (8e)
\end{align*}
$$

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 $\mathcal{D}_{\text{data}}$ 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” [12]) form. Accordingly, it holds that

$$
\begin{align*}
x(k+1) &= (A - KC)x(k) + Bu(k) + Ky(k) \\
e(k) &= y(k) - Cx(k) + Du(k),
\end{align*}
$$

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

$$
x(t) = (A - KC)^\rho x(t-\rho) + \sum_{p=1}^{\rho} \Phi_p u(t-p) + \Psi_p y(t-p),
$$

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

$$
x(t) = C u_{\mathcal{I}} + O(|\lambda_{\text{max}}|^\rho) \rightarrow 0 \quad \text{as} \quad \rho \rightarrow \infty
$$

where the $O(\cdot)$ term goes to zero exponentially; $C$ stacks the (reversed) controllability matrices $C_u$ and $C_y$, i.e.,

$$
C = \begin{bmatrix} C_u & C_y \end{bmatrix} = \begin{bmatrix} \Phi_\rho & \cdots & \Phi_2 & \Phi_1 & \Psi_\rho & \cdots & \Psi_2 & \Psi_1 \end{bmatrix},
$$

and

$$
u_{\mathcal{I}} := \begin{bmatrix} u(t-\rho) \\ \vdots \\ u(t-2) \\ u(t-1) \end{bmatrix}, \quad y_{\mathcal{I}} := \begin{bmatrix} y(t-\rho) \\ \vdots \\ y(t-2) \\ y(t-1) \end{bmatrix}
$$

are noisy collections of past inputs and outputs. The relation in (11) thus guarantees that, up to $O(|\lambda_{\text{max}}|^\rho)$ 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, $\exists C_{\text{det}}$ such that

$$
x(t) = C_{\text{det}} u_{\mathcal{I}} + \Psi_{\mathcal{I}} y_{\mathcal{I}},
$$

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

**Remark 2 (Choice of $\rho$ - part I)** In (subspace) system identification, see e.g., [4,11,10,13] the quantity $\rho$, known also as the “past horizon”, has to be determined from measured data trading off bias and variance. Indeed, $\rho$ should be large, so that the quantity $O(|\lambda_{\text{max}}|^\rho)$ can be neglected, but a large $\rho$ ultimately requires estimating larger sample covariance matrices. A simple and effective way of determining $\rho$ in a data-driven fashion is by using Akaike’s criterion (e.g., FPE) [1], with the latter choice also guaranteeing that $\| (A - KC)^\rho \| = O(|\lambda_{\text{max}}|^\rho) = O_p(1/\sqrt{\mathcal{N}_{\text{data}}})$. This is in contrast with common practice in the literature of DDPC where the length $\rho$ 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 $\rho$ 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 $D_{\text{data}}$. The results in this section are standard in subspace identification and can be found in several references, see for instance [5,16,4,9,11,10].

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” $\rho$ and “future” $T$ are fixed, the number of columns $N$ of the Hankel data matrices is chosen such a way that all the available data are exploited, namely $N = N_{\text{data}} - T - \rho$.

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 $H_d \in \mathbb{R}^{pT \times mT}$ and $H_s \in \mathbb{R}^{pT \times pT}$ formed with its Markov parameters, i.e.,

$$H_d = \begin{bmatrix} D & 0 & 0 & \ldots & 0 \\ CB & D & 0 & \ldots & 0 \\ CAB & CB & D & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CAT^{-2}B & CAT^{-3}B & CAT^{-4}B & \ldots & D \end{bmatrix}, \quad (16a)$$

and

$$H_s = \begin{bmatrix} I & 0 & 0 & \ldots & 0 \\ CK & I & 0 & \ldots & 0 \\ CAK & CK & I & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CAT^{-2}K & CAT^{-3}K & CAT^{-4}K & \ldots & I \end{bmatrix}. \quad (16b)$$

Based on (11) and provided $\rho$ is chosen in a data-driven fashion as discussed in Remark 2, $X_\rho$ can be written as

$$X_\rho = C_u U_P + C_y Y_P + (A - K C)^\rho X_0 + CZ_P, \quad (17)$$

where $Z_P = [U_P^T \ Y_P]^T$. The Hankel matrix of future outputs $Y_F$ thus satisfies the following:

$$Y_F = \Gamma X_\rho + H_d U_F + H_s E_F + CZ_P + H_d U_F + H_s E_F, \quad (18)$$

which is the equation often considered as a starting point in subspace identification and control [32,12,18]. We can now characterize the future noise $E_F$ according to the following.

Lemma 1 (Projection of noise) For any fixed $\rho$ in (12), it holds that

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

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

Proof By definition,

$$\Pi_{Z_{p,F}}(E_F) = E_F \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} Z_P & Z_P \\ U_F & U_F \end{bmatrix}^{-1} \begin{bmatrix} Z_P \\ U_F \end{bmatrix},$$

so that

$$\Upsilon = E_F \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} Z_P & Z_P \\ U_F & U_F \end{bmatrix}^{-1} \begin{bmatrix} Z_P \\ U_F \end{bmatrix}. \quad \left[\Sigma_{\rho,F} \Sigma_{\rho,F} \right]$$

It is sufficient to observe that the term on the left-hand side $\Sigma_{\rho,F}$ and $\Sigma_{\rho,F}$ are sample cross-covariances between future innovations and past data $(z_P)_{p,F}$ 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.
Theorem 1 (Output/data relation) Given any \( \alpha \in \mathbb{R}^N \), the vector \( \hat{y}_f^\alpha := \hat{Y}_F\alpha \) satisfies the relation

\[
\hat{y}_f^\alpha = \Gamma \hat{x}^d(t) + H_d u_f + O_p \left( \frac{1}{\sqrt{N}} \right),
\]

where \( \hat{x}^d(t) := \hat{X}_\rho \alpha = CZ_P\alpha = C_{z_{init}} \),

\[
\begin{bmatrix}
  u(t) \\
  u(t+1) \\
  \vdots \\
  u(t+T-1)
\end{bmatrix} = U_F\alpha, \quad (22b)
\]

and \( z_{init} := Z_P\alpha \).

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

\[
\hat{y}_f^\alpha := \hat{Y}_F\alpha \triangleq \Gamma C Z_P\alpha + H_d U_F\alpha.
\]

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.,

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

is a linear transformation through \( \alpha \) 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 \( \text{Var}\{e(t)\} > 0 \), 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.

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

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

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 \( z(t) := [u^T(t) y^T(t)]^T \) 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. [26]). 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.

\footnote{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.}
Theorem 2 (Output/data relation - generalized)
Under the assumptions in Lemma 3, given any (past) joint input and output trajectory
\[ z_{\text{init}} := \begin{bmatrix} z(t - \rho) \\ \vdots \\ z(t - 2) \\ z(t - 1) \end{bmatrix}, \tag{24} \]
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
\[ y_f^d := \begin{bmatrix} y_f^d(t) \\ y_f^d(t + 1) \\ \vdots \\ y_f^d(t + T - 1) \end{bmatrix} \]
satisfies:
\[ y_f^d = \hat{Y}_F \alpha^* + O_P(1/\sqrt{N}) \triangleq \hat{Y}_F \alpha^* \tag{26} \]
where \( \alpha^* \) is the minimum-norm solution of the system of linear equations:
\[ \begin{bmatrix} z_{\text{init}} \\ u_f \end{bmatrix} = \begin{bmatrix} Z_p \\ U_F \end{bmatrix} \alpha \tag{27} \]
where \( \hat{Y}_F := \Pi_{Z_p, U_F}(Y_F) \).

Proof Under the assumption of Lemma 3, the matrix \( Z_{\text{data}} \) has full rank and, therefore, \( \forall z_{\text{init}} \) and \( u_f \), there exists \( \alpha \) such that
\[ \begin{bmatrix} z_{\text{init}} \\ u_f \end{bmatrix} = \begin{bmatrix} Z_p \\ U_F \end{bmatrix} \alpha. \tag{28} \]
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 \( \alpha^* \).

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 for \( \rho > n \), it is well know (see, e.g., [30]) that the Hankel matrix \( Z_{\text{data}} \) in (23) and \( Z_P \in \mathbb{R}^{(m+p)\rho \times N} \) in (27) have rank equal to
\[ \text{rank}(Z_{\text{data}}) = n + m(\rho + T) < (m + p)(\rho + T), \]
\[ \text{rank}(Z_P) = n + mp < (m + p)\rho. \]
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” [34]. Nonetheless, in this case, any finite (deterministic) trajectory \( z_{\text{init}} \) of the system (6) belongs to the column span of \( Z_P \). As such, provided that \( z_{\text{init}} \) 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:

\[ \begin{array}{l}
\text{minimize}_{u_f} \ J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) \\
\text{s.t.} \quad \alpha^* = \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} z_{\text{init}} \\ u_f \end{bmatrix}, \\
y_f^d = \hat{Y}_F \alpha^*, \\
u(k) \in \mathcal{U}, \quad y_f^d(k) \in \mathcal{Y}, \quad k \in [t, t + T),
\end{array} \tag{29} \]

where
\[ J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) = \frac{1}{2} \left[ \sum_{k=t}^{t+T-1} ||y_f^d(k) - y_r||_Q^2 + ||u(k) - u_r||_R^2 \right], \tag{29e} \]
and \( \hat{Y}_F := \Pi_{Z_p, U_F}(Y_F) \), while \( z_{\text{init}} \) 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 Stochastic Predictive Control (SPC) problem firstly formalized in [18].

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 [19, Section IV.B], the formulation tailored to cope with bounded noise introduced in [7] and that with elastic net regularization given in [17, Section IV.D].

4.1 SPC with slacks

Based on our notation, the SPC problem tackled in [19] to handle non-deterministic scenarios (with measurement noise only) can be recast as follows:

\[
\text{minimize } J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) + \lambda \| \sigma \|_2^2 \quad (30a)
\]

s.t. \[ \alpha = \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \begin{bmatrix} z_{init} \\ u_f \end{bmatrix} + \begin{bmatrix} \sigma \\ 0 \end{bmatrix}, \quad (30b) \]
\[ y_f^d = Y_P \alpha, \quad (30c) \]
\[ u(k) \in U, \quad y_f^d(k) \in \mathcal{Y}, \quad k \in [t, t + T), \quad (30d) \]

where \( \sigma \in \mathbb{R}^{p(m+p)} \) 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 \( \sigma \) 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 \( \rho \) 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 \( \rho \) and \( N_{data} \) and the value of the slack variables needed to counteract the effect of noise on the initial conditions. Indeed, when \( \rho \) is selected according to Akaike’s criterion, Remark 2 directly links the accuracy of the reconstructed dimension with the size of the dataset. As such, for large \( N_{data} \), then \( \lambda \to +\infty \)

\( \lambda \) is the optimal choice. This will be confirmed by the simulation results reported in Section 6 (see Figure 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 \( \| e(t) \|_\infty \leq \varepsilon \), and \( \varepsilon \) is assumed to be known. In our framework, the regularized problem proposed in [7] to tackle this scenario can be rewritten as follows:

\[
\text{minimize } J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) + \lambda \varepsilon \| \alpha \|_2^2 + \lambda \varepsilon \| \sigma \|_2^2 \quad (31a)
\]

s.t. \[ \begin{bmatrix} z_{init} + I_y \sigma_{init} \\ u_f \\ y_f^d + \sigma_y \end{bmatrix} = \begin{bmatrix} Z_P \\ U_F \end{bmatrix} \alpha, \quad (31b) \]
\[ u(k) \in U, \quad y_f^d(k) \in \mathcal{Y}, \quad k \in [t, t + T), \quad (31c) \]
\[ \| \sigma \|_\infty \leq \varepsilon (1 + \| \alpha \|_1), \quad k \in [t, t + T), \quad (31d) \]
\[ \sigma = \begin{bmatrix} \sigma_{init} \\ \sigma_y \end{bmatrix} \in \mathbb{R}^{p(T+p)}, \quad (31f) \]

where \( \sigma \) is a vector of slacks accounting for the noise acting on the output measurements used to build \( Z_P \) and \( Y_P \), while \( I_y \) 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 constraint that connects the slack variables to the known features of the measurement noise.

By recasting the robust DDPC formulation in [7] 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 [7] complemented with an additional constraint on the slack variable \( \sigma_y \) 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 (29c) 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

\( \Pi \) is the projection matrix used to project the output measurements on the known features of the measurement noise.
\( \Pi \) is the orthogonal projector onto the column span of 
\[ \begin{bmatrix} Z_P^T & U_F^T \end{bmatrix}, \]
i.e.,
\[ \Pi := \begin{bmatrix} Z_P^T & \bar{U}_F^T \end{bmatrix}, \] (32)

the solutions of (29) with terminal constraints and (31) coincide for \( \lambda, \bar{\varepsilon} = 0 \) and \( \lambda \to \infty \).

**Proof** For \( \lambda \to \infty \) and \( \lambda, \bar{\varepsilon} = 0 \), (31) reduces to

\[
\begin{align*}
\text{minimize} & \quad J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) , \\
\text{s.t.} & \quad \begin{bmatrix} \bar{z}_{\text{init}} + \mathbf{y}_{\text{init}}
\\
y_f^d + \mathbf{y}_f
\\
\end{bmatrix} = Z_P \begin{bmatrix} U_F \end{bmatrix} \mathbf{\alpha}, \\
& \quad \begin{bmatrix} \mathbf{y}_f \end{bmatrix} = Y_F \mathbf{\alpha},
\end{align*}
\]

(33a)

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

\[
\begin{align*}
\bar{z}_{\text{init}} + \mathbf{y}_{\text{init}} & = Z_P \begin{bmatrix} U_F \end{bmatrix} \mathbf{\alpha}, \\
y_f^d + \mathbf{y}_f & = Y_F \mathbf{\alpha},
\end{align*}
\]

(34)

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

\[ \mathbf{\sigma}_f = Y_F(I - \Pi) \mathbf{\alpha} = 0, \]

(35)

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

\[
\begin{align*}
\mathbf{\alpha} & = \begin{bmatrix} Z_P^T \\
U_F \end{bmatrix}^\dagger \begin{bmatrix} \bar{z}_{\text{init}} \\
n_{\text{init}} \end{bmatrix} , \\
y_f^d = Y_F \mathbf{\alpha} = Y_F \Pi \mathbf{\alpha} + Y_F(I - \Pi) \mathbf{\alpha} = Y_F \Pi \mathbf{\alpha}.
\end{align*}
\]

(36)

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

The relationship established in Theorem 3 are consistent with the empirical evidences on the role of \( \lambda, \bar{\varepsilon} \) and \( \lambda \), discussed in [7, Section V]. Although providing a guideline for the choice of these two hyperparameters, with the choice of \( \lambda \) inherently connected with the noise bound \( \bar{\varepsilon} \), it is worth stressing that this choice will be optimal asymptotically, i.e., when \( \rho \) is selected according to the Akaike’s criterion and \( N_{\text{data}} \to \infty \).

### 4.3 DeePC with elastic net regularization

We now consider the problem with elastic net regularization in [17, Section IV.D], that we rewrite for the control problem considered in this work by using our notation as follows:

\[
\begin{align*}
\text{minimize} & \quad J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) + \lambda_1 \| \mathbf{\alpha} \|_1 + \lambda_2 \|(I - \Pi) \mathbf{\alpha} \|_p \quad \text{(38a)} \\
\text{s.t.} & \quad \begin{bmatrix} z_{\text{init}} \\
y_f^d \end{bmatrix} = Z_P \begin{bmatrix} U_F \end{bmatrix} \mathbf{\alpha}, \\
& \quad \begin{bmatrix} \mathbf{y}_f \end{bmatrix} = Y_F \mathbf{\alpha}, \\
& \quad u(k) \in \mathcal{U}, \quad y^d(k) \in \mathcal{Y}, \; k \in [t, t + T),
\end{align*}
\]

(38b)

where \( \Pi \) 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 (29c), 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

\[
\begin{align*}
\text{minimize} & \quad J \left( \begin{bmatrix} y_f^d \\ u_f \end{bmatrix} \right) , \\
\text{s.t.} & \quad \begin{bmatrix} z_{\text{init}} \\
y_f^d \end{bmatrix} = Z_P \begin{bmatrix} U_F \end{bmatrix} \mathbf{\alpha}, \\
& \quad \begin{bmatrix} \mathbf{y}_f \end{bmatrix} = Y_F \mathbf{\alpha}, \\
& \quad u(k) \in \mathcal{U}, \quad y^d(k) \in \mathcal{Y}, \; k \in [t, t + T) .
\end{align*}
\]

(39a)

In addition, by decomposing \( Y_F := \bar{Y}_F + \bar{\mathcal{V}}_F \), where \( \bar{Y}_F = \Pi Z_P, U_F(Y_F) \), and \( \bar{\mathcal{V}}_F = Y_F - \Pi Z_P, U_F(Y_F) \), we have that

\[ \bar{Y}_F = Y_F \Pi \quad \bar{\mathcal{V}}_F = Y_F(I - \Pi). \]

Then, when \( (I - \Pi) \mathbf{\alpha} = 0 \), we have

\[ Y_F \mathbf{\alpha} = Y_F \Pi \mathbf{\alpha} + Y_F(I - \Pi) \mathbf{\alpha} = Y_F \Pi \mathbf{\alpha} = \bar{Y}_F \mathbf{\alpha}. \]
experimentally, into a rigorous frame. Note that, since the performance of SPC is influenced by the choice of \( \rho \) 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 \( \rho \) is chosen according to the Akaike’s criterion and \( N_{\text{data}} \to \infty \) (i.e., \( N \to \infty \)).

5 The \( \gamma \)-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 \( Z\text{data} \) in (23), namely:

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

where the matrices \( \{L_{ij}\}_{i,j=1}^3 \) 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:

\[
\begin{align*}
\gamma_{\text{init}} &= Z_P \alpha = L_{11} Q_1 \alpha \tag{42a} \\
\gamma_f &= U_F \alpha = \begin{bmatrix}
L_{21} & L_{22}
\end{bmatrix}
\begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix} \alpha, \tag{42b}
\end{align*}
\]

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

\[
\hat{y}_f = \hat{Y}_F \alpha = \begin{bmatrix}
L_{31} & L_{32}
\end{bmatrix}
\begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix} \alpha^*, \tag{43}
\]

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

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

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

\[
\begin{align*}
\gamma_{\text{init}} &= L_{11} \gamma_1^*, \\
\alpha^* &\text{ can be written as } \alpha^* = Q_1^\top \gamma_1^*, \text{ so that } \\
\alpha^* &\in \text{colspan} (Q_1^\top).
\end{align*}
\]

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

\[
Q_1 \alpha = L_{11}^{-1} \gamma_{\text{init}} = \gamma_1^*.
\]

The minimum-norm solution \( \alpha^* \) can be found by as

\[
\alpha^* = Q_1^\top \gamma_1^*,
\]

thus concluding the proof.

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

\[
\begin{align*}
\gamma_f &= L_{21} \gamma_1^* + L_{22} \gamma_2 \\
\gamma_2 &= Q_2 \alpha.
\end{align*}
\]

Based on this representation, we can provide additional insights on \( \alpha^* \), through the following result.

**Lemma 6 (Definition of \( \gamma_2^* \))** Let \( \alpha^* \in \mathbb{R}^N \) indicate the minimum-norm \( \alpha \) solving (45). Accordingly, define \( \gamma_2^* \in \mathbb{R}^{m\rho} \) as the unique solution of the least squares problem

\[
L_{22} \gamma_2 = \gamma_f - L_{21} \gamma_1^* \tag{46}
\]

where \( \gamma_1^* \) is defined in Lemma 5. Then \( \alpha^*_f \) can be written as:

\[
\alpha^*_f = Q_2^\top \gamma_2^*, \tag{47}
\]

so that

\[
\alpha^*_f \in \text{colspan} (Q_2^\top).
\]

**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 \( \alpha \) to

\[
\begin{align*}
\gamma_f &= L_{21} \gamma_1^* + L_{22} \gamma_2 \\
\gamma_2 &= Q_2 \alpha.
\end{align*}
\]

must therefore satisfy

\[
Q_2 \alpha = L_{22}^{-1} [\gamma_f - L_{21} \gamma_1^*],
\]

and the minimum-norm solution is given by

\[
\alpha^*_f = Q_2^\top L_{22}^{-1} [\gamma_f - L_{21} \gamma_1^*] = Q_2^\top L_{22}^{-1} [\gamma_f - L_{21} \gamma_1^*] = Q_2^\top \gamma_2^*.
\]
From Lemma 5 and 6, we can then characterize the minimum-norm parameter $\alpha$ of the whole behavioral model in (29b)-(29c) as follows.

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

1. $\alpha^*_\text{init} = Q_1^\top \gamma_1^*; 
2. $\alpha^*_j = Q_2^\top \gamma_2^*; 
3. $\alpha^*_\text{init}$ is orthogonal to $\alpha^*_j; 
4. $Q_1 \alpha_1^* = 0$ and $Q_2 \alpha^*_\text{init} = 0; 
5. $Q_3 \alpha_3^* = Q_3 \alpha^*_\text{init} = 0.

Therefore, $\alpha^* = \alpha^*_\text{init} + \alpha^*_j$ is the minimum-norm vector satisfying the conditions:

\[
\begin{bmatrix}
Z_f \\
U_f
\end{bmatrix}
\begin{bmatrix}
\alpha^* \\
\gamma^*
\end{bmatrix}
= 
\begin{bmatrix}
L_{11} & 0 \\
L_{21} & L_{22}
\end{bmatrix}
\begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix}
\begin{bmatrix}
\gamma_1^* \\
\gamma_2^*
\end{bmatrix},
\]

(48a)

\[
g_f^d = \sum_{i=1}^3 L_{3i} Q_i \alpha_i^* = 2 \sum_{i=1}^2 L_{3i} Q_i \gamma_i^* = 2 \sum_{i=1}^2 L_{3i} \gamma_i^*. 
\]

(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, $\alpha^* = \alpha^*_\text{init} + \alpha^*_j$ is a solution to (48). The fact that it is the minimum-norm solution derives from the fact that $\alpha^*$ belongs to the column space of $[Q_1^\top, Q_2^\top]$.

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

\[
\text{minimize}_{\gamma_1, \gamma_2} J \left( \begin{bmatrix} g_f^d \\ u_f \end{bmatrix} \right), 
\]

(49a)

\[
\begin{bmatrix}
z_f^\text{init} \\
L_{11} & 0 \\
L_{21} & L_{22} \\
L_{31} & L_{32}
\end{bmatrix}
\begin{bmatrix}
\gamma_1 \\
\gamma_2
\end{bmatrix},
\]

(49b)

\[
\begin{bmatrix} u_f \\ y_f 
\end{bmatrix} = 
\begin{bmatrix}
L_{21} & L_{22} \\
L_{31} & L_{32}
\end{bmatrix}
\begin{bmatrix}
\gamma_1^* \\
\gamma_2^*
\end{bmatrix},
\]

(49c)

\[
u(k) \in \mathcal{U}, y_f(k) \in \mathcal{Y}, k \in [t, t + T),
\]

where the cost is defined in (29c) and we reshape the predictor based on the properties of the minimum-norm $\alpha$ 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 $\gamma_1$. In turn, $\gamma_1$ is not a proper optimization variable, but acts as a constraint that can be explicitly solved by setting:

\[
\gamma_1^* = L_{11}^{-1} z^\text{init}. 
\]

(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 to Lemma 5, $\gamma_1^*$ and $\gamma_2$ found through (50) coincides with the ones leading to the minimum-norm $\alpha$ satisfying the initial conditions.

The constrained optimization problem to be solved at each time instant thus result into a reduced problem on $\gamma_2$ only, i.e.,

\[
\text{minimize}_{\gamma_2} J \left( \begin{bmatrix} g_f^d \\ u_f \end{bmatrix} \right), 
\]

(51a)

\[
\begin{bmatrix} u_f \\ y_f 
\end{bmatrix} = 
\begin{bmatrix}
L_{21} & L_{22} \\
L_{31} & L_{32}
\end{bmatrix}
\begin{bmatrix}
\gamma_1^* \\
\gamma_2
\end{bmatrix},
\]

(51b)

\[
u(k) \in \mathcal{U}, y_f(k) \in \mathcal{Y}, k \in [t, t + T),
\]

(51c)

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

According to this decomposition, we propose the $\gamma$-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 $\gamma$-DDPC scheme is likely to be computationally advantageous. Indeed, the dimension of the optimization variable $\gamma_2 \in \mathbb{R}^{m_T}$ in (51) is likely to be considerably smaller than the one of $\alpha \in \mathbb{R}^N$. At the same time, retrieving $\gamma_1$ requires the inversion of a matrix with dimensions dictated by the chosen $\rho$.

**Remark 4 (Choice of $\rho$ (part II))** The length of the “past” window plays a pivotal role in shaping the performance of the predictive controller. On the one hand, $\rho$ should be chosen by following an identification-oriented reasoning (see Remark 2). On the other, a smaller $\rho$ reduces the number of data needed to solve the DDPC problem ($N_{\text{data}} := N + T + \rho$), 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 $\alpha$ 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 $\alpha$ by introducing a regularizer in the cost, as already proposed in [17,7]. Although such procedure has proven to be effective, the regularization strength has to be well calibrated to trade-off between reducing the norm of $\alpha$ 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 $\gamma$-DDPC scheme, while validating the conclusions drawn in Section 5.1, we consider the same benchmark example proposed in [6]. Therefore, the unknown plant to be controlled is described by the following model:

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

(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 problem by running...
Fig. 4. Closed-loop validation tests (SNR = 18 dB): absolute differences between the performance indexes of \( \gamma \)-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 (SNR = 18 dB): absolute differences between the performance indexes of \( \gamma \)-DDPC and the average values of those associated with the noisy oracle MPC vs length of the “past horizon” \( \rho \) over 30 Monte Carlo predictors.

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

\[
J = \sum_{t=0}^{T_v-1} \| y(t) \|_Q^2 + \| u(t) \|_R^2, \quad J_u = \sum_{t=0}^{T_v-1} u(t)^2,
\]

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 \( \gamma \)-DDPC scheme introduced in Section 5. Firstly, we assess the sensitivity of \( \gamma \)-DDPC to noise in the available batch of data \( D_{N_{data}} \). 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 \( \gamma \)-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 \( \gamma \)-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 \( \gamma \)-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 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.

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 thus highlight the possible advantages of using the \( \gamma \)-DDPC scheme over a identification+model-based control procedure, at least for the considered case study. We also evaluate how \( \gamma \)-DDPC performs when increasing \( N_{data} \) over noisy

---

\( ^6 \) All tests have been carried out on an M1 chip, running MATLAB 2021a, while the optimization problems are solved with CVX [24,25].

\( ^7 \) The model is identified with N4SID [33].

\( ^8 \) The average order of the ARMAX models is 7, while its standard deviation is 5.
Fig. 6. Closed-loop validation tests (SNR = 18 dB): absolute differences between the performance indexes of $\gamma$-DDPC and the average values of those associated with the noisy oracle MPC vs penalties $\beta$ on a 2-norm regularization on $\gamma_2$ over 30 Monte Carlo predictors.

6.1 Effect of additional regularization on $\gamma$-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 $\gamma_2$, 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 $\beta$. However, when $\beta$ 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 $\gamma_3$ not set to zero beforehand, as in the proposed $\gamma$-DDPC approach. In this case, $\gamma_3$ 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 $\gamma_3$ we obtain performance comparable with the ones obtained with the oracle predictive controller. Specifically low $\eta$ 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
Fig. 8. Closed-loop validation tests (SNR = 18 dB): absolute differences between the performance indexes attained with [19] and the average values of those associated with the noisy oracle MPC vs penalties $\lambda$ over 30 Monte Carlo predictors.

Fig. 9. Closed-loop validation tests (SNR = 18 dB): absolute differences between the performance indexes attained with [7] and the average values of those associated with the noisy oracle MPC vs penalties $\bar{\lambda}_\alpha$ and $\lambda_\sigma$ over 30 Monte Carlo predictors.

Fig. 10. Closed-loop validation tests (SNR = 18 dB): absolute differences between the average performance indexes attained with the noisy oracle MPC and $\mathcal{J}$ and $\mathcal{J}_u$ obtained with $\gamma$-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.

The choice of the regularization parameters is crucial to attain satisfactory performance when exploiting the approach proposed in [7], 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 $\lambda_\sigma$ 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 $\alpha$, along with introducing a set of slacks, requires a careful selection of both the associated the regularization penalty. When compared with $\gamma$-DDPC, even with the best possible tuning, the schemes presented in [19] and [7] result in the worst average performance with respect to the oracle MPC and a higher variability of the closed-loop behavior, see Figure 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, Fig. 11 and Fig. 12 corroborate the conclusions drawn in Section 4. Indeed, it is clear that larger values of $\lambda_2$ and smaller $\lambda_1$ lead to performance that are comparable with that of the $\gamma$-DDPC. In particular, for $\lambda_1 = 10^{-8}$, it is clear that $\lambda_2$ has an
Fig. 11. Closed-loop validation tests (SNR = 18 dB): absolute differences between the performance indexes attained with [17] and the average values of those associated with the noisy oracle MPC vs penalties $\lambda_1$ and $\lambda_2$ over 5 Monte Carlo predictors.

Fig. 12. Average inputs for closed-loop validation tests (SNR = 18 dB): oracle (dashed black line) vs average input (line) and standard deviation (shaded area) for different values of $\lambda_1$ and $\lambda_2$ in (38) over 5 Monte Carlo predictors.

7 Conclusions

In this paper, exploiting subspace identification tools, we have provided an unifying framework for several regularized data-driven predictive control schemes proposed in the literature, 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 $\gamma$-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 $\gamma$-DDPC ease 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 $\gamma$-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

[1] H. Akaike. Fitting autoregressive models for prediction. Annals of the Institute of Statistical Mathematics, 21:243–247, 1969.
[2] Amr Alanwar, Yvonne Stürz, and Karl Henrik Johansson. Robust data-driven predictive control using reachability analysis. arXiv preprint arXiv:2103.14110, 2021.
[3] Daniele Alpago, Florian Dörfler, and John Lygeros. An extended kalman filter for data-enabled predictive control. IEEE Control Systems Letters, 4(4):994–999, 2020.
[4] D. Bauer. Asymptotic properties of subspace estimators. Automatica, 41:359–376, 2005.
[5] D. Bauer and M. Jansson. Analysis of the asymptotic properties of the MOESP type of subspace algorithms. Automatica, 36:497–509, 2000.
[6] Alberto Bemporad, Manfred Morari, Vivek Dua, and Estratos N Pistikopoulos. The explicit linear quadratic regulator for constrained systems. Automatica, 38(1):3–20, 2002.
[7] Julian Berberich, Johannes Köhler, Matthias A Müller, and Frank Allgöwer. Data-driven model predictive control with stability and robustness guarantees. IEEE Transactions on Automatic Control, 66(4):1702–1717, 2020.
[8] Valentina Breschi and Simone Formentin. Direct data-driven design of switching controllers. International Journal of Robust and Nonlinear Control, 30(15):6042–6072, 2020.
[9] A. Chiuso. Asymptotic variance of closed-loop subspace identification algorithms. *IEEE Trans. on Aut. Control*, 51(8):1299–1314, 2006.

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

[11] A. Chiuso. The role of Vector AutoRegressive modeling in predictor based subspace identification. *Automatica*, 43(6):1034–1048, June 2007.

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

[13] Alessandro Chiuso. 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, 2010.

[14] Jeremy Coulson, John Lygeros, and Florian Dörfler. Data-enabled predictive control: In the shallows of the deepc. In 2019 18th European Control Conference (ECC), pages 307–312. IEEE, 2019.

[15] Jeremy Coulson, John Lygeros, and Florian Dörfler. Regularized and distributionally robust data-enabled predictive control. In 2019 IEEE 58th Conference on Decision and Control (CDC), pages 2696–2701, 2019.

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

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

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

[19] Félix Fiedler and Sergio Lucia. On the relationship between data-enabled predictive control and subspace predictive control. In 2021 *European Control Conference (ECC)*, pages 222–229, 2021.

[20] Simone Formentin, Marco C Campi, Algo Carè, and Sergio M Savaresi. Deterministic continuous-time virtual reference feedback tuning (vrft) with application to pid design. *Systems & Control Letters*, 127:25–34, 2019.

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

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

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

[24] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

[25] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. *http://cvxr.com/cvx*, Mar 2014.