

Data-Enabled Predictive Iterative Control

Kai Zhang, Riccardo Zuliani, Efe C. Balta, and John Lygeros

Abstract—This work introduces the Data-Enabled Predictive iteRative Control (DeePRC) algorithm, a direct data-driven approach for iterative LTI systems. The DeePRC learns from previous iterations to improve its performance and achieves the optimal cost. By utilizing a tube-based variation of the DeePRC scheme, we propose a two-stage approach that enables safe active exploration using a left-kernel-based input disturbance design. This method generates informative trajectories to enrich the historical data, which extends the maximum achievable prediction horizon and leads to faster iteration convergence. In addition, we present an end-to-end formulation of the two-stage approach, integrating the disturbance design procedure into the planning phase. We showcase the effectiveness of the proposed algorithms on a numerical experiment.

Index Terms—Data-driven control, Iterative learning control, Model predictive control, Active exploration.

I. INTRODUCTION

Direct data-driven control enables decision-making directly from raw data, bypassing the need for a parametric model, which received significant attention in the recent literature. The Fundamental Lemma [1] establishes that all finite-length trajectories of a linear time-invariant (LTI) system belong to the span of the Hankel matrix constructed using a suitably persistently exciting input/output trajectory, and has inspired numerous novel controller designs in the framework of behavioral systems theory. For example, [2] focuses on robustly stabilizing data-driven feedback control design, while [3], [4] develop a predictive control scheme in a receding-horizon manner. The celebrated data-enabled predictive control (DeePC) scheme of [3] enables novel applications and theories in the field of direct data-driven control. Later extensions of DeePC provide also closed-loop stability guarantees by enforcing the last segment of the predicted trajectory to match the target [4].

The DeePC scheme uses a Hankel matrix representation of the system dynamics, which is assumed to satisfy a *persistence of excitation condition* (PE condition) [1]. The degree of excitation of the Hankel matrix limits the maximum prediction horizon of the controller; hence, a lack of sufficiently rich historical input/output data may produce short prediction horizons, potentially leading to suboptimal performance. Our approach is to use the measurement data obtained during system execution to improve the PE condition, thus allowing for longer prediction horizons. This requires persistently exciting inputs which need careful design to ensure exploration

without sacrificing performance. We address this problem in the context of iterative systems by providing a novel scheme with active exploration and theoretical guarantees.

Iterative systems are those performing repeated tasks, where all individual tasks, called iterations, start from the same initial condition and share the same objective. This enables the possibility of iteratively updating the controller using data collected from each iteration. In the context of iterative systems, [5] proposes a learning Model Predictive Control (LMPC) algorithm that constructs a safe set using past safe trajectories to be used as terminal constraint. This strategy ensures asymptotic stability and a non-increasing cost over iterations, ultimately converging to the optimal cost under suitable assumptions. To improve robustness, [6] extends LMPC to systems with additive uncertainties by leveraging tube-MPC approaches [7]. In this case, robust safe sets are constructed using nominal trajectories satisfying tightened constraints to ensure the nominal cost is non-increasing over iterations.

In this paper, we present the Data-Enabled Predictive iteRative Control (DeePRC), a direct data-driven control approach for iterative LTI systems. The DeePRC uses an input/output convex safe set and a terminal cost function designed from previous trajectories and enjoys recursive feasibility, asymptotic stability, non-increasing iteration cost, and convergence to the infinite-horizon optimum, albeit asymptotically. We develop an active exploration variant of DeePRC, which is able to increase its maximum prediction horizon in a sample-efficient way requiring *only a single initial safe trajectory* through the design of an input disturbance term. We further present an end-to-end formulation of the two-stage approach, integrating the disturbance design procedure into the planning phase.

Notation: $[a, b]$ is the set of integers a to b . $\|x\|_p$ is the p -norm of x and $\|x\|_A^2 = x^\top Ax$. $\text{col}(x_0, x_1, \dots, x_\ell)$ and $\text{col}(A_0, A_1, \dots, A_\ell)$ denote vertically stacked vector $[x_0^\top \ x_1^\top \ \dots \ x_\ell^\top]^\top$ and matrix $[A_0^\top \ A_1^\top \ \dots \ A_\ell^\top]^\top$. For a matrix A , $A_{[m,:]}$ denotes the submatrix of its first m rows, while $A_{[m,:]}$ represents the remaining rows and we use $A_{[-m,:]}$ to denote its last m rows. $\text{image}(A)$ and $\text{leftker}(A)$ denote the image and left kernel of matrix A , respectively. For a generic length- N sequence $\{y_k\}_{k=0}^{N-1}$, we write $y_{[a,b]} := \text{col}(y_a, \dots, y_b)$.

II. PRELIMINARIES

A. Problem Setting

Consider the unknown discrete-time LTI system

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t, \\ y_t &= Cx_t + Du_t, \end{aligned} \quad (1)$$

where $x_t \in \mathbb{R}^n$ is the state, $u_t \in \mathbb{R}^m$ is the input, and $y_t \in \mathbb{R}^p$ is the output. The state x_t is not directly measurable, and needs to be reconstructed using input/output

information. Denote by $\underline{\ell} \in \mathbb{N}_{>0}$ the lag of the system, that is, the minimum integer such that the observability matrix $\mathcal{O}_{\underline{\ell}}(A, C) := \text{col}(C, CA, \dots, CA^{\underline{\ell}-1})$ has rank n . We assume that an upper bound ℓ on the lag $\underline{\ell}$ is available.

To represent the state using input/output data, we define the extended state following [8, Definition 3]:

$$\xi_t := \text{col}(u_{[t-\ell, t-1]}, y_{[t-\ell, t-1]}) \in \mathbb{R}^{n_\xi}, \quad (2)$$

where $n_\xi = (m+p)\ell$. Here, ξ_t is the state of a non-minimal representation of (1) with dynamics

$$\begin{aligned} \tilde{\xi}_{t+1} &= \tilde{A}\xi_t + \tilde{B}u_t, \\ y_t &= \tilde{C}\xi_t + \tilde{D}u_t, \end{aligned} \quad (3)$$

which is generally not unique. We refer to [8, Equation 6] for one possible form of (3).

Suppose system (1) performs iterative tasks, where the task for each iteration is to steer the system from a starting equilibrium ξ^S to a target equilibrium ξ^F . We define $u^j = \text{col}(u_{-\ell}^j, \dots, u_{-1}^j, u_0^j, \dots)$ and $y^j = \text{col}(y_{-\ell}^j, \dots, y_{-1}^j, y_0^j, \dots)$, where $u_t^j = u^S$, $y_t^j = y^S$, $\forall t \leq 0$ describe the initial state for the associated extended state sequence $\{\xi_t^j\}_{t=0}^{T^j}$, with $\xi_0^j = \xi^S$.

A trajectory is considered *safe* if it converges to the target state $\xi^F := \text{col}(u^F, \dots, u^F, y^F, \dots, y^F)$ and satisfies the convex input and output constraints

$$u_t^j \in \mathcal{U}, \quad y_t^j \in \mathcal{Y}, \quad (4)$$

for all $t \in \mathbb{Z}_{[0, T^j-1]}$, where $\mathcal{U} \subseteq \mathbb{R}^m$ and $\mathcal{Y} \subseteq \mathbb{R}^p$ are convex sets. Following [5], the control objective is to solve the following infinite horizon problem.

$$J_\infty^*(\xi^S) = \min_{u_0, u_1, \dots} \sum_{k=0}^{\infty} h(y_k, u_k) \quad (5a)$$

$$\text{s.t.} \quad \xi_{k+1} = \tilde{A}\xi_k + \tilde{B}u_k, \quad \forall k \geq 0, \quad (5b)$$

$$y_k = \tilde{C}\xi_k + \tilde{D}u_k, \quad \forall k \geq 0, \quad (5c)$$

$$\xi_0 = \xi^S, \quad (5d)$$

$$u_k \in \mathcal{U}, \quad y_k \in \mathcal{Y}, \quad \forall k \geq 0. \quad (5e)$$

The stage cost is $h(u_k, y_k) = \|u_k - u^F\|_R^2 + \|y_k - y^F\|_Q^2$ with $Q, R \succ 0$, and we assume $(u^F, y^F) \in \text{int}(\mathcal{U} \times \mathcal{Y})$.

Assumption 1. *System (1) is observable and stabilizable.*

Problem (5) is not tractable due to the infinite horizon and the lack of knowledge of the system dynamics. We develop the DeePRC algorithm as a tractable solution to (5). Our scheme uses a data-based non-parametric model for (5b)-(5c) to perform receding horizon control.

B. Review of DeePC

Here, we provide a brief background on the Data Enabled Predictive Control (DeePC). Let $T, L \in \mathbb{Z}_{\geq 0}$ with $T \geq L > \ell$. The Hankel matrix of depth L for a signal $u = \text{col}(u_0, \dots, u_T)$ is defined as

$$\mathcal{H}_L(u) := \begin{bmatrix} u_0 & u_1 & \cdots & u_{T-L} \\ u_1 & u_2 & \cdots & u_{T-L+1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{L-1} & u_L & \cdots & u_{T-1} \end{bmatrix}.$$

We use the identifiability condition presented in [9] for theoretical guarantees, which is a slight generalization of the Fundamental Lemma [1].

Lemma 1 ([9, Corollary 21]). *Consider a length T input/output trajectory $\{u^d, y^d\}$ of system (1). The image of $\mathcal{H}_L(u^d, y^d) := \text{col}(\mathcal{H}_L(u^d), \mathcal{H}_L(y^d))$ is the span of all length L trajectories of the system if and only if*

$$\text{rank}(\mathcal{H}_L(u^d, y^d)) = mL + n. \quad (6)$$

Under condition (6), one could identify a state-space (using standard system identification techniques) or kernel representation (following [10]) of the system. The main advantage of a direct data-driven approach, like the one presented in this paper, is its simplicity and efficiency, as well as the possibility to directly deal with noisy data by introducing regularization terms in the cost function. For a detailed discussion about the differences between direct and indirect methods, we refer the reader to [11]. A direct comparison between our approach and that of [10] is subject of future research. Given an initial horizon $T_{\text{ini}} \in \mathbb{N}_{>0}$ and a prediction horizon $N \in \mathbb{N}_{>0}$, let

$$\begin{bmatrix} U_P \\ U_F \end{bmatrix} := \mathcal{H}_{T_{\text{ini}}+N}(u^d), \quad \begin{bmatrix} Y_P \\ Y_F \end{bmatrix} := \mathcal{H}_{T_{\text{ini}}+N}(y^d), \quad (7)$$

where $U_P \in \mathbb{R}^{mT_{\text{ini}} \times H}$, $U_F \in \mathbb{R}^{mN \times H}$, $Y_P \in \mathbb{R}^{pT_{\text{ini}} \times H}$, $Y_F \in \mathbb{R}^{pN \times H}$ with $H = T - T_{\text{ini}} - N + 1$. Suppose that (u^d, y^d) satisfy the rank condition (6) with $L = T_{\text{ini}} + N$. According to Lemma 1, $\{\text{col}(u_{\text{ini}}, u), \text{col}(y_{\text{ini}}, y)\}$ is a valid trajectory of the system if and only if $\exists g \in \mathbb{R}^H$ such that

$$\begin{bmatrix} U_P \\ Y_P \\ U_F \\ Y_F \end{bmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}, \quad (8)$$

where $u_{\text{ini}} \in \mathbb{R}^{mT_{\text{ini}}}$ and $y_{\text{ini}} \in \mathbb{R}^{pT_{\text{ini}}}$ are the T_{ini} -step past input/output observations. If $T_{\text{ini}} \geq \underline{\ell}$, given any future control input sequence $u \in \mathbb{R}^{mN}$, the associated output sequence $y \in \mathbb{R}^{pN}$ is uniquely determined via (8) thanks to Lemma 1. In the remainder of this paper, we assume $T_{\text{ini}} = \ell$.

III. NOMINAL DEEPRC

In this section, we introduce the nominal DeePRC algorithm and exploration schemes using past trajectory data.

A. Input/Output Convex Safe Set

The input/output convex safe set ioCS^j at iteration j is defined as the convex hull of all *extended states* along previous safe trajectories:

$$\text{ioCS}^j := \left\{ \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^i \xi_t^i \mid \gamma_t^i \geq 0, \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^i = 1 \right\}, \quad (9)$$

where $M^j := \{i \in \mathbb{Z}_{[0, j-1]} \mid \sum_{t=0}^{\infty} h(y_t^i, u_t^i) < \infty, u_t^i \in \mathcal{U}, y_t^i \in \mathcal{Y}, \forall t\}$ indicates past safe iterations. Finite trajectories can be extended by appending (y^F, u^F) . For all $i \leq j$, $M^i \subseteq M^j$, and therefore $\text{ioCS}^i \subseteq \text{ioCS}^j$. Every $\xi \in \text{ioCS}^j$ is safe, meaning that it can be steered to ξ^F without violating

the constraints. Note that ioCS^j is updated only in between iterations. The *cost-to-go* of ξ_t^i is defined as

$$J^i(\xi_t^i) = \sum_{k=t}^{\infty} h(u_k^i, y_k^i).$$

representing the cost required to steer ξ_t^i to ξ^F . Note that $J^i(\xi_0^i) = J^i(\xi^S)$ is the cost of the entire trajectory i . The cost-to-go can be extended to any $\xi \in \text{ioCS}^j$, by defining

$$\begin{aligned} P^j(\xi) = \min_{\gamma \geq 0} & \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^i J^i(\xi_t^i) \\ \text{s.t.} & \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^i = 1, \quad \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^i \xi_t^i = \xi. \end{aligned} \quad (10)$$

Note that $P^j(\xi)$ represents the cost required to safely steer ξ to ξ^F under the control policy $\pi(\xi) = \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^{i,*} u_t^i$, where γ^* is the minimizer of (10).

Theorem 2. *The set ioCS^j is control invariant¹ for all j . For any $\xi \in \text{ioCS}^j$, $P^j(\xi^+) \leq P^j(\xi) - h(\bar{u}, \bar{y})$, $\bar{u} \in \mathcal{U}$ and $\bar{y} \in \mathcal{Y}$, where $\xi^+ = \tilde{A}\xi + \tilde{B}\bar{u}$, $\bar{y} = \tilde{C}\xi + \tilde{D}\bar{u}$, and $\bar{u} = \pi(\xi)$.*

Proof. For any $\xi \in \text{ioCS}^j$, let

$$\bar{u} := \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^{i,*} u_t^i, \quad \bar{y} := \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^{i,*} y_t^i,$$

where $\bar{u} \in \mathcal{U}$ and $\bar{y} \in \mathcal{Y}$ because of convexity, and γ^* is a minimizer of (10). By linearity we have $\xi^+ := \tilde{A}\xi + \tilde{B}\bar{u} \in \text{ioCS}^j$ and, recognizing that $\tilde{\gamma}$, defined for $t \geq 1$ as $\tilde{\gamma}_t^i = \gamma_{t-1}^{i,*}$, is a feasible solution of (10) with $\xi = \xi^+$, we have

$$\begin{aligned} P^j(\xi^+) &\leq \sum_{i \in M^j} \sum_{t=0}^{\infty} \tilde{\gamma}_{t+1}^i J^i(\xi_{t+1}^i) \\ &= \sum_{i \in M^j} \sum_{t=0}^{\infty} \gamma_t^{i,*} (J^i(\xi_t^i) - h(u_t^i, y_t^i)) \\ &\leq P^j(\xi) - h(\bar{u}, \bar{y}), \end{aligned}$$

where the last inequality follows from convexity. \square

B. The Nominal DeePRC Algorithm

The nominal DeePRC algorithm utilizes a single safe trajectory $\{u^0, y^0\}$ to construct both the Hankel matrix and the safe set ioCS^1 . Since the closed-loop trajectories can potentially have infinite length, we assume only a finite portion of them is used to construct the non parametric representation of the system. We use T^j to denote the length of the recorded trajectory.

Assumption 2. *A safe trajectory $\{u^0, y^0\}$ is available with $u_{[0, T^0]}^0, y_{[0, T^0]}^0$ satisfying (6) with $L = T_{\text{ini}} + N$ and $N \geq \ell$.*

Let $\text{col}(U_P, Y_P, U_F, Y_F) = \mathcal{H}_{T_{\text{ini}}+N}(u^0, y^0)$, $u(t) \in \mathbb{R}^{mN}$, $y(t) \in \mathbb{R}^{pN}$, $g(t) \in \mathbb{R}^H$, and

$$J_N^j(u(t), y(t), \xi_N(t)) := \sum_{k=0}^{N-1} h(u_k(t), y_k(t)) + P^j(\xi_N(t)).$$

¹We refer the reader to [12] for an exact definition.

The nominal DeePRC for time-step t and iteration j is formulated as follows:

$$J_N^{j,*}(\xi_t^j) = \min_{g(t), u(t), y(t)} J_N^j(u(t), y(t), \xi_N(t)) \quad (11a)$$

$$\text{s.t.} \quad \begin{bmatrix} U_P \\ Y_P \\ U_F \\ Y_F \end{bmatrix} g(t) = \begin{bmatrix} u_{\text{ini}}(t) \\ y_{\text{ini}}(t) \\ u(t) \\ y(t) \end{bmatrix}, \quad \begin{bmatrix} u_{\text{ini}}(t) \\ y_{\text{ini}}(t) \end{bmatrix} = \xi_t^j, \quad (11b)$$

$$\xi_N(t) = \begin{bmatrix} u_{[N-\ell, N-1]}(t) \\ y_{[N-\ell, N-1]}(t) \end{bmatrix}, \quad \xi_N(t) \in \text{ioCS}^j, \quad (11c)$$

$$u_k(t) \in \mathcal{U}, \quad y_k(t) \in \mathcal{Y}, \quad \forall k \in [0, N-1], \quad (11d)$$

C. Hankel Matrix Update

The DeePRC (11) utilizes a static Hankel matrix. As new iterations $\{u^j, y^j\}$ are measured, we can augment the prediction horizon by modifying the Hankel matrix. At iteration j , the largest feasible prediction horizon is

$$N^j = \max \left\{ N \mid \text{rank}(\mathcal{H}_N^j) = m(\ell + N) + n \right\}, \quad (12)$$

with $\mathcal{H}_N^j = [\mathcal{H}_{\ell+N}(u^0, y^0) \quad \dots \quad \mathcal{H}_{\ell+N}(u^{j-1}, y^{j-1})]$. Suppose our goal is to have $N^j = \bar{N} \geq \ell$ for some j . The desired horizon \bar{N} is a design parameter that can be chosen before the beginning of the operation, or can be incrementally updated online until some stopping criterion is met. We assume that $\bar{N} < T^j$ for all j . To fulfill the objective we require $\mathcal{H}_{\bar{N}}^j$ to satisfy the rank condition (6) with $L = \bar{L} := \ell + \bar{N}$. If $\bar{N} > N^j$, then we solve (11) with prediction horizon N^j and matrix $\mathcal{H}_{N^j}^j$. The process continues until $\text{rank}(\mathcal{H}_{\bar{N}}^j) = m\bar{L} + n$ for some \bar{j} , after which we initialize (11) with \bar{N} and $\mathcal{H}_{\bar{N}}^{\bar{j}+1}$.

This type of update takes place at the end of each iteration, however, we can also study how each new sample, gathered within the iteration, affects the rank of $\mathcal{H}_{\bar{N}}^j$. To this end, let $\mathcal{H}_{\bar{N}}^{j,t} = \mathcal{H}_{\bar{N}}^j$ for all $t < \bar{N} - 1$ and define for all $t \geq \bar{N} - 1$

$$\mathcal{H}_{\bar{N}}^{j,t+1} = \begin{bmatrix} \mathcal{H}_{\bar{N}}^{j,t} & \begin{bmatrix} u_{[t-\bar{L}+1, t]}^j \\ y_{[t-\bar{L}+1, t]}^j \end{bmatrix} \end{bmatrix}. \quad (13)$$

Ideally, each update in (13) should ensure that $\text{rank} \mathcal{H}_{\bar{N}}^{j,t+1} = 1 + \text{rank} \mathcal{H}_{\bar{N}}^{j,t}$. Note that the matrices $\mathcal{H}_{\bar{N}}^{j,t}$ obtained through (13) are not used in the DeePRC problem, they are only used to understand how newly gathered samples affect the rank of the Hankel matrix of depth \bar{N} , and will later be used for the input disturbance design.

IV. SAFE AND ACTIVE EXPLORATION

Update (13) can be inefficient, as new trajectories are not guaranteed to increase the rank condition and extend the prediction horizon. In this section, we present an *active exploration* strategy to safely reach the desired prediction horizon \bar{N} . We design the control input as the sum of two components $u_t = \tilde{u}_t + d_t$, where \tilde{u}_t is chosen to optimize closed-loop performance, and d_t is an artificial input disturbance ensuring active exploration.

A. Safe Exploration: Tube DeePRC

To ensure safety, we assume d_t is chosen from a known, user-designed bounded set \mathcal{D} , and develop a tube-based DeePRC scheme given by

$$\min_{g(t), v(t), z(t)} J_N^j(v(t), z(t), \zeta_N(t)) \quad (14a)$$

$$\text{s.t. } (g(t), v(t), z(t), \zeta_N(t)) \in \mathcal{C}, \quad (14b)$$

$$v_k(t) \in \bar{\mathcal{U}}, z_k(t) \in \bar{\mathcal{Y}}, \forall k \in [0, N-1], \quad (14c)$$

where \mathcal{C} denotes the constraints (11b)-(11c) with (u, y, ξ) replaced with (v, z, ζ) , and z_t, v_t , and ζ_t are the nominal output, input, and extended state, respectively. The tightened constraints $\bar{\mathcal{U}} = \mathcal{U} \ominus \mathcal{E}_u$ and $\bar{\mathcal{Y}} = \mathcal{Y} \ominus \mathcal{E}_y$ are obtained using a robust positive invariant set \mathcal{E} by $\mathcal{E}_u = \{e_u \in \mathbb{R}^m \mid \exists e \in \mathcal{E}, \text{ s.t. } e_u = T_u e\}$ and $\mathcal{E}_y = \{e_y \in \mathbb{R}^p \mid \exists e \in \mathcal{E}, \text{ s.t. } e_y = T_y e\}$, with T_u and T_y selecting the entries associated to the last input and the last output of the error e between the true and the nominal extended state. We then choose

$$\tilde{u}_t^j = K(\xi_t^j - \zeta_t^j) + v_0^*(t), \quad (15)$$

where K is a stabilizing output feedback gain (which can be computed either using [2, Theorem 8] or, under less restrictive conditions, using the techniques of [13]).

One way to obtain \mathcal{E} is to consider the nominal system

$$\begin{aligned} \zeta_{t+1} &= \tilde{A}\zeta_t + \tilde{B}v_t \\ z_t &= \tilde{C}\zeta_t + \tilde{D}v_t. \end{aligned}$$

and obtain (\tilde{A}, \tilde{B}) from (8) with $N = 1$ as

$$\xi_{t+1} = \begin{bmatrix} \text{col}(U_{\mathcal{P}[m:, :]}, U_{\mathcal{F}}) \\ \text{col}(Y_{\mathcal{P}[p:, :]}, Y_{\mathcal{F}}) \end{bmatrix} \begin{bmatrix} U_{\mathcal{P}} \\ Y_{\mathcal{P}} \\ U_{\mathcal{F}} \end{bmatrix}^\dagger \begin{bmatrix} u_{\text{ini}}(t) \\ y_{\text{ini}}(t) \\ u(t) \end{bmatrix} = \tilde{A}\xi_t + \tilde{B}u_t,$$

which always admits a solution under Assumption 2 thanks to [2, Theorem 1]. Assuming (\tilde{A}, \tilde{B}) is stabilizable (sufficient conditions for stabilizability have been studied in [2, Theorem 3], alternatively, one can perform a change of coordinates to obtain the extended-state of [13]) we can choose \mathcal{E} (either polytopic or ellipsoidal) to satisfy

$$e_{t+1} = (\tilde{A} + \tilde{B}K)e_t + \tilde{B}d_t \in \mathcal{E}, \forall e_t \in \mathcal{E}, \forall d_t \in \mathcal{D}, \quad (16)$$

where $e_t := \xi_t - \zeta_t$. One practical way to avoid the (\tilde{A}, \tilde{B}) parameterization, at the expense of a potential violation of the theoretical guarantees, is to choose \mathcal{E} to be a sufficiently large ball centered at the origin. Data-driven techniques to estimate \mathcal{E} are an active area of research and represent an interesting direction for future work.

For safe planning, any element in the ioCS of problem (1) is required to meet the tightened constraints $\bar{\mathcal{U}}, \bar{\mathcal{Y}}$. As in [6], we need to strengthen Assumption 2 as follows.

Assumption 3. A robustly safe trajectory $\{v^0, z^0\}$ (i.e., with $v_t^0 \in \bar{\mathcal{U}}, z_t^0 \in \bar{\mathcal{Y}}, \forall t \in \mathbb{Z}_{\geq 0}$) is available with $v_{[0, T^0]}^0, z_{[0, T^0]}^0$ satisfying (6) with $L = T_{\text{ini}} + N$ and $N \geq \ell$.

To ensure robust constraint satisfaction, we construct the ioCS in (14) using the nominal trajectories $\{v^j, z^j\}$.

B. Active Exploration: LKB Input Disturbance Design

Since \tilde{u}_t^j in (15) satisfies the nominal constraints for all $d_t^j \in \mathcal{D}$ (compare Corollary 4), we can design d_t^j to ensure the Hankel matrix update (13) always produces an increment in the rank of the matrix. Choosing d_t^j independently and uniformly from \mathcal{D} at each time step may be inefficient, since the control input \tilde{u}_t^j may already contain sufficient information to excite the system. We propose a left-kernel-based (LKB) disturbance design inspired by [14] as a sample-efficient exploration strategy.

For the first $N^j - 2$ time-steps of iteration j we set $d_t^j = 0$. Next, for $t \geq N^j - 1$, let $\text{col}(\bar{U}^{j,t}, \bar{Y}^{j,t}) := \mathcal{H}_{\bar{N}}^{j,t}$, where $\bar{U}^{j,t} \in \mathbb{R}^{m\bar{L} \times H_c}$, $\bar{Y}^{j,t} \in \mathbb{R}^{p\bar{L} \times H_c}$ with $H_c = t - \bar{N} + 1 + \sum_{i=0}^{j-1} (T^i - \bar{N} + 1)$, and let $\bar{Y}_{\text{up}}^{j,t} = \bar{Y}_{[p(\bar{L}-1), :]}^{j,t}$. If

$$\text{col}(y_{[t-\bar{L}+1, t-1]}^j, u_{[t-\bar{L}+1, t-1]}^j, \tilde{u}_t^j) \notin \text{image} \begin{bmatrix} \bar{Y}_{\text{up}}^{j,t} \\ \bar{U}^{j,t} \end{bmatrix}, \quad (17)$$

then the candidate control input \tilde{u}_t^j will excite the trajectory and thus we set $d_t^j = 0$. Otherwise, consider any $\kappa \in \mathbb{R}^{p(\bar{L}-1) + m\bar{L}}$ in the left kernel of $\text{col}(\bar{Y}_{\text{up}}^{j,t}, \bar{U}^{j,t})$ with $\kappa_{[-m:]} \neq 0$ and take any d_t^j satisfying

$$\kappa^\top \text{col}(y_{[t-\bar{L}+1, t-1]}^j, u_{[t-\bar{L}+1, t-1]}^j, \tilde{u}_t^j + d_t^j) \neq 0, \quad (18)$$

where $\kappa_{[-m:]}$ denotes the last m elements of κ [14]. If \mathcal{D} is absorbing (e.g. any norm ball), there always exists some d_t^j satisfying (18). We then apply the input $u_t^j = \tilde{u}_t^j + d_t^j$ with \tilde{u}_t^j as in (15). This ensures that $\text{rank}(\mathcal{H}_{\bar{N}}^{j,t+1}) = \text{rank}(\mathcal{H}_{\bar{N}}^{j,t}) + 1$ for all $t \geq N^j - 1$, meaning that the desired horizon \bar{N} can be reached in exactly $m\bar{L} + n - \text{rank}(\mathcal{H}_{\bar{N}}^1)$ time steps. Afterwards, $d_t^j \equiv 0$ and the nominal DeePRC (11) can be used instead of (14). The DeePRC scheme with LKB active exploration is summarized in Algorithm 1.

C. An End-to-End Formulation

Algorithm 1 presents a two-stage approach for designing the control input. We also provide an end-to-end formulation, which integrates the online disturbance design procedure into the optimization problem

$$\min_{g(t), v(t), z(t), d(t)} J_N^j(v(t), z(t), \zeta_N(t)) + \lambda \|d(t)\|_1 \quad (19a)$$

$$\text{s.t. } (g(t), v(t), z(t), \zeta_N(t)) \in \mathcal{C}, \quad (19b)$$

$$v_k(t) \in \bar{\mathcal{U}}, z_k(t) \in \bar{\mathcal{Y}}, \forall k \in [0, N-1], \quad (19c)$$

$$u_0(t) = v_0(t) + K(\xi_t^j - \zeta_t^j), \quad (19d)$$

$$\left\| \begin{bmatrix} y_{[t-\bar{L}+1, t-1]}^j \\ u_{[t-\bar{L}+1, t-1]}^j \\ u_0(t) + d(t) \end{bmatrix}^\top \mathcal{K}^{j,t} \right\|_\infty \geq \epsilon, d(t) \in \mathcal{D}. \quad (19e)$$

Condition (17) is here enforced through the excitation constraint (19e), where $\epsilon \in \mathbb{R}_{>0}$ and $\mathcal{K}^{j,t}$ is an orthogonal basis of $\text{col}(\bar{Y}_{\text{up}}^{j,t}, \bar{U}^{j,t})$. We further add 1-norm penalty on the disturbance to ensure that its magnitude is kept to a minimum. (19) can be formulated as a mixed-integer quadratic program (MIQP). The end-to-end formulation should generally perform better than the two-stage, as $d(t)$ is part of the optimization problem. Similar to the two-stage approach, we only solve (19) until \bar{N} is reached, then we utilize (11).

Algorithm 1 DeePRC with LKB Active Exploration

```

1: Init:  $\{u^0, y^0\}, \xi^S, \xi^F, \bar{N} \geq \ell, \mathcal{D}, \text{ioCS}^1, j = 1.$ 
2: while  $y^j \neq y^{j-1}$  do
3:    $t = 0, \xi_0^j = \xi^S, \zeta_0^j = \xi^S.$ 
4:   if  $\bar{N}$  has been reached then
5:     Setup (11) using  $\mathcal{H}_{\bar{N}}^{j+1}.$ 
6:     while not converged to  $\xi^F$  do
7:       | Solve (11) and apply  $u_t^j.$  Get  $y_t^j$  and  $\xi_{t+1}^j.$ 
8:     end while
9:      $\text{ioCS}^{j+1} = \text{ioCS}^j \cup \{u^j, y^j\}.$ 
10:  else
11:    Compute  $N^j$  using (12).
12:    Setup (14) using  $\mathcal{H}_{N^j}^j.$ 
13:    while not converged to  $\xi^F$  do
14:      | Solve (14) and get  $\tilde{u}_t^j$  from (15).
15:      if  $t \leq \bar{N} - 2$  or (17) holds then
16:        |  $d_t^j = 0.$ 
17:      else
18:        | Select any  $d_t^j$  satisfying (18).
19:        | Update  $\mathcal{H}_{\bar{N}}^{j,t}$  with (13).
20:      end if
21:      | Apply  $u_t^j = \tilde{u}_t^j + d_t^j.$  Get  $y_t^j$  and  $\xi_{t+1}^j.$ 
22:    end while
23:     $\text{ioCS}^{j+1} = \text{ioCS}^j \cup \{v^j, z^j\}.$ 
24:  end if
25:   $j \leftarrow j + 1.$ 
26: end while

```

V. PROPERTIES AND GUARANTEES

A. Closed-Loop Properties

We first prove that the scheme in (11) enjoys recursive feasibility and stability. We require the following.

Assumption 4. *There exists $c_u > 0$ such that $J_{\bar{N}}^{j,*}(\xi) \leq c_u \|\xi\|_2^2$ for all ξ for which (11) is feasible.*

Proposition 3. *Under Assumptions 1, 2, and 4, the following hold for all $j \geq 1$: i) Problem (11) is feasible for all $t \in \mathbb{Z}_{\geq 0}$; ii) (u^j, y^j) satisfy the constraints \mathcal{U}, \mathcal{Y} ; iii) ξ_t^j exponentially converges to ξ^F .*

Proof. Assumption 2 ensures $\xi_\tau^0 \in \text{ioCS}^j, \forall \tau \in [0, T^0 - 1]$. At $t = 0$, as $\xi_0^j = \xi^S = \xi_0^0$, the length- N trajectory $\hat{u}(0) = u_{[0, N-1]}^0, \hat{y}(0) = y_{[0, N-1]}^0$ taken from $\{u^0, y^0\}$ is a feasible solution. Next, suppose $u^*(t), y^*(t), g^*(t)$ is the solution of (11) at time step t . At time step $t + 1$, let $\hat{u}(t + 1) = \text{col}(u_{[1, N-1]}^*(t), \bar{u}), \hat{y}(t + 1) = \text{col}(y_{[1, N-1]}^*(t), \bar{y})$, where $\bar{u} = \pi(\xi_{\bar{N}}^*(t)) \in \mathcal{U}$. According to Theorem 2, $(\hat{u}(t + 1), \hat{y}(t + 1))$ satisfies all constraints and there exists some $\hat{g}(t + 1)$ satisfying (11b) since $(\hat{u}(t + 1), \hat{y}(t + 1))$ is a valid trajectory of the system. This proves recursive feasibility. Using the candidate solution $\hat{u}(t + 1), \hat{y}(t + 1)$ and Theorem 2, we can prove exponential stability by following the arguments in [4]. \square

As the Tube DeePRC (14) solves a similar problem to the nominal DeePRC, it inherits similar closed-loop properties.

Corollary 4. *Under Under Assumptions 1, 3, and 4 (with ξ replaced with ζ and (11) replaced with (14)) the following holds for all $j \geq 1$: i) Problem (14) is feasible for all $t \in \mathbb{Z}_{\geq 0}$; ii) (v^j, z^j) satisfy the tightened constraints $\bar{\mathcal{U}}, \bar{\mathcal{Y}}$, and ζ_t^j exponentially converges to ξ^F ; iii) (u^j, y^j) satisfy the nominal constraints \mathcal{U}, \mathcal{Y} , and $\xi_t^j \rightarrow \xi^F + \mathcal{E}$ for all $d_t^j \in \mathcal{D}$.*

B. Recursive Feasibility of the End-to-End Formulation

Let $\mathcal{D} = \{d \in \mathbb{R}^m \mid \|d\|_\infty \leq \bar{d}\}$ for some $\bar{d} \in \mathbb{R}_{>0}$, and let $\mathcal{K}_{\bar{u}}^{j,t} = \mathcal{K}_{[-m, :, :]}^{j,t}$.

Proposition 5. *Under Assumption 3, if $\bar{d} \geq \epsilon$ and the 1-norm of each non-zero column of $\mathcal{K}_{\bar{u}}^{j,t}$ is 1, then (19) is recursively feasible.*

Proof. Let $\mathcal{K}_{\bar{u}}^{j,t}$ denote the i -th column of $\mathcal{K}_{\bar{u}}^{j,t}$, and suppose it is non-zero. Given any $\hat{u}_0(t)$, we have

$$\begin{aligned} & \|\text{col}(y_{[t-\bar{L}+1, t-1]}^j, u_{[t-\bar{L}+1, t-1]}^j, \hat{u}_0(t) + \hat{d}(t))^\top \mathcal{K}_{\bar{u}}^{j,t}\|_\infty \\ &= \|c + \hat{d}(t)^\top \mathcal{K}_{\bar{u}}^{j,t}\|_\infty \geq |c_i + \hat{d}(t)^\top \mathcal{K}_{\bar{u}}^{j,t}| \end{aligned} \quad (20)$$

where c is a constant row vector and c_i denotes its i -th element. By choosing $\hat{d}(t) = \bar{d} \cdot \text{sign}(\mathcal{K}_{\bar{u}}^{j,t})$ if $c_i \geq 0$ and $\hat{d}(t) = -\bar{d} \cdot \text{sign}(\mathcal{K}_{\bar{u}}^{j,t})$ otherwise, with $\text{sign}(\cdot)$ is applied element-wise, it holds that (20) $\geq \bar{d} \cdot \|\mathcal{K}_{\bar{u}}^{j,t}\|_1 \geq \epsilon$. In other words, for any $\hat{u}_0(t)$, we can construct the disturbance $\hat{d}(t)$ such that the excitation constraint (19e) is satisfied. The rest of the proof constructs a candidate input/output sequence similar to that of Proposition 3, such that constraints (19b)-(19c) are satisfied. \square

C. Convergence Properties

From the discussion in Subsection IV-B, we immediately have the following.

Proposition 6. *Both the DeePRC with LKB active exploration in Algorithm 1 and the end-to-end formulation (19) reach the desired prediction horizon \bar{N} in exactly $m\bar{L} + n - \text{rank}(\mathcal{H}_{\bar{N}}^1)$ time steps.*

Proposition 6 implies that we can obtain finite horizon approximations of (5) in finitely many time steps and a direct data-driven fashion. Moreover, if ξ_t in (5) reaches the origin in finitely many time-steps c , and $c \leq \bar{N}$, then (u^j, y^j) converges to the solution of (5) in finitely many iterations. Using the arguments in [15], we further have the following.

Theorem 7. *Let Assumption 2 hold, and suppose the desired prediction horizon \bar{N} is achieved at iteration \bar{j} . Then $J^j(\xi^S) \geq J^{\bar{j}+1}(\xi^S)$ for all $j > \bar{j}$. Moreover, if $u^{\bar{j}+1} = u^{\bar{j}}$ in Algorithm 1 for some finite \bar{j} and the LICQ condition in [15, Assumption 3] holds, then $u^{\bar{j}}$ solves (5).*

Proof. The first point follows from [5, Theorem 2]. The second point follows from [15, Theorem 2]. \square

For $j \leq \bar{j}$, we can also prove that the nominal cost of (14) is non-increasing over iterations. Theorem 7 guarantees that the solution of (5) be reached asymptotically for any horizon N . However, controllers with larger horizons generally lead to

faster convergence, as showcased in the simulations of Section VI, and exhibit overall better performance.

VI. NUMERICAL EXAMPLES

We consider the following four-dimensional system

$$A = \begin{bmatrix} 1 & 0 & 0.1 & 0 \\ 0 & 1 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \\ 0 & 0.1 \\ 0.1 & 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

with $D = 0$, subject to $\mathcal{U} = [-1.5, 1.5]^2$, $\mathcal{Y} = [-1.5, 1.5]^2$. We assume the true lag $\ell = 2$ is unknown and set $\ell = 4$. The initial and target states are $(u^S, y^S) = ([0 \ 0]^\top, [0 \ 0]^\top)$ and $(u^F, y^F) = ([0 \ 0]^\top, [0.4 \ -0.4]^\top)$. The cost matrices are $Q = \mathbf{I}^4$ and $R = 0.1 \cdot \mathbf{I}^2$. In this setting, the desired prediction horizon is $\bar{N} = 50$. The initial safe trajectory $\{u^0, y^0\}$ is obtained using the LQR controller ($\bar{Q} = \mathbf{I}^4, \bar{R} = \mathbf{I}^2$). To meet Assumption 3, small random disturbances are added to the LQR control input for the first 20 time steps. Using (12), we verify that $\{u^0, y^0\}$ produces a prediction horizon of $N^1 = 8$.

In Figure 1, we compare the performance of passive Hankel update (Passive), two-stage with LKB design (2s-LKB), end-to-end approach (end-to-end). The figure illustrates the relationship between the rank of the matrix $\mathcal{H}_{\bar{N}}^j$ and its number of columns. Without exploration, the rank increment is considerably slower, indicating that many added trajectories to the matrix lack sufficient information. Meanwhile, for the methods with exploration, the trend is approximately linear, as expected. In Table I we further compare the closed-loop costs of the 2-stage LKB and the 1-stage approaches against a nominal DeePRC scheme with fixed horizon $N = 8$ (without exploration) and against the passive scheme. The methods with active exploration achieve worse cost in $j = 1$ due to constraint tightening, but outperform the nominal and the passive schemes later thanks to the extended prediction horizon. In the first iteration, the end-to-end performs better than the LKB, since disturbance design is incorporated in the MPC problem, however, it requires an average computation

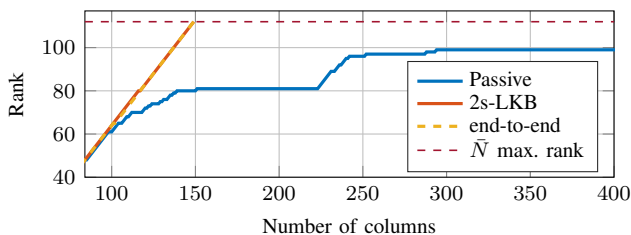


Fig. 1. Evolution of the rank of the Hankel matrix.

TABLE I
PREDICTION HORIZON AND TRAJECTORY COSTS OVER ITERATIONS

j	Nominal		Passive		2s-LKB		End-to-end	
	N	J^j	N	J^j	N	J^j	N	J^j
0	\	8.635158	\	8.635158	\	8.635158	\	8.635158
1	8	7.784081	8	7.818932	8	7.827334	8	7.826908
2	8	7.765810	17	7.756467	50	7.748497	50	7.748497
3	8	7.761973	19	7.749598	50	7.748497	50	7.748497
4	8	7.760775	20	7.748698	50	7.748497	50	7.748497

time of 0.757s, as opposed to the 0.058 seconds of the two-stage methods. The difference in performance between active and passive methods in our example is small, highlighting that further research is required to understand the impact of prediction horizon on closed-loop performance. Indeed, Theorem 3 guarantees that the desired prediction horizon will be reached through active exploration, but provides no claims about the relative performance improvement.

VII. CONCLUSION

In this work, we presented the DeePRC algorithm, a direct data-driven approach to control LTI systems performing iterative tasks. The DeePRC requires only an initial safe trajectory with a low excitation order and it guarantees sample-efficient rank-increment while maintaining safety in closed-loop. Under suitable conditions, DeePRC is guaranteed to converge to the best achievable performance. Extending the formal guarantees for systems with process disturbances, measurement noise, nonlinearities, and non-identical initial conditions are promising directions for future work.

REFERENCES

- [1] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. De Moor, "A note on persistency of excitation," *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, Apr. 2005.
- [2] C. De Persis and P. Tesi, "Formulas for Data-Driven Control: Stabilization, Optimality, and Robustness," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 909–924, Mar. 2020.
- [3] J. Coulson, J. Lygeros, and F. Dörfler, "Data-Enabled Predictive Control: In the Shallows of the DeePC," in *2019 18th European Control Conference (ECC)*, June 2019, pp. 307–312.
- [4] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-Driven Model Predictive Control With Stability and Robustness Guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, Apr. 2021.
- [5] U. Rosolia and F. Borrelli, "Learning Model Predictive Control for Iterative Tasks. A Data-Driven Control Framework," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1883–1896, July 2018.
- [6] U. Rosolia, X. Zhang, and F. Borrelli, "Robust learning model-predictive control for linear systems performing iterative tasks," *IEEE Transactions on Automatic Control*, vol. 67, no. 2, pp. 856–869, 2021.
- [7] D. Q. Mayne, M. M. Seron, and S. V. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, Feb. 2005.
- [8] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "On the design of terminal ingredients for data-driven MPC," *IFAC-PapersOnLine*, vol. 54, no. 6, pp. 257–263, Jan. 2021.
- [9] I. Markovsky and F. Dörfler, "Identifiability in the Behavioral Setting," *IEEE Transactions on Automatic Control*, vol. 68, no. 3, pp. 1667–1677, Mar. 2023.
- [10] M. Alsalti, M. Barkey, V. G. Lopez, and M. A. Müller, "Sample-and-computationally efficient data-driven predictive control," *arXiv preprint arXiv:2309.11238*, 2023.
- [11] F. Dörfler, J. Coulson, and I. Markovsky, "Bridging Direct and Indirect Data-Driven Control Formulations via Regularizations and Relaxations," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 883–897, Feb. 2023.
- [12] F. Borrelli, *Constrained optimal control of linear and hybrid systems*, ser. Lecture notes in control and information sciences. Berlin Heidelberg: Springer, 2003, no. 290.
- [13] M. Alsalti, V. G. Lopez, and M. A. Müller, "Notes on data-driven output-feedback control of linear mimo systems," *arXiv preprint arXiv:2311.17484*, 2023.
- [14] H. J. Van Waarde, "Beyond Persistent Excitation: Online Experiment Design for Data-Driven Modeling and Control," *IEEE Control Systems Letters*, vol. 6, pp. 319–324, 2022.
- [15] U. Rosolia, Y. Lian, E. T. Maddalena, G. Ferrari-Trecate, and C. N. Jones, "On the Optimality and Convergence Properties of the Iterative Learning Model Predictive Controller," *IEEE Transactions on Automatic Control*, vol. 68, no. 1, pp. 556–563, Jan. 2023.