Closed-loop data-enabled predictive control

Jack Umenberger

***Abstract*— Recent years have witnessed renewed interest in so-called data-enabled control, in which predictions of system behavior are made directly from measured data, in place of an explicit system model. In this paper, we consider systems with measurable disturbances, and present a closed-loop data enabled predictive control strategy that searches over sequences of feedback policies, thereby extending the open-loop strategy recently proposed in Huang et al. [1]. We make use of recent advances in polytopic containment to derive convex formulations of the minimax (i.e. worst-case disturbances) control problem, for a variety of cost functions, constraints, and sets of allowable disturbances. Advantages over open-loop and non-robust alternatives are illustrated with a numerical example.**

I. INTRODUCTION

Model-based control has been a dominant paradigm in theory and practice since, at least, the breakthrough contributions [2], [3] of Kalman in the early 1960s. Model-based control, of course, requires knowledge of an accurate model of the system. In some settings, such models can be constructed from first principles; increasingly, however, models must be 'learned', or at least refined, with observed data, in a process called 'system identification' [4].

Due, perhaps in part, to an increasing abundance and dependence on data, as well as the dramatic success of reinforcement learning in games [5], [6], recent years have witnessed a resurgence in research activity at the intersection of learning from data and control, cf. [7] for a recent survey. In particular, there has been renewed interest in so-called 'data-enabled control', cf. *Related work* below. The data-enabled approach eschews the use of an explicit system model; instead, predictions about system behavior are made directly from observed data, under the assumption that this behavior can be well-approximated by a linear dynamical model. Such an approach is not so much 'model-free' as it is 'model-implicit'; quantities associated with the implicit model, such as parameter coefficients or state dimension, are no longer explicitly specified.

Recent work [1] has extended the data-enabled approach to systems with disturbances, under the assumption that past - but not future - disturbances are directly measurable by the controller; cf. §II-B for a detailed problem specification. In the present paper, we build upon [1], offering the following two principal contributions. First, we present a closed-loop formulation of the data-enabled predictive control problem, in which one searches over a sequence of feedback policies, rather than an open-loop sequence of inputs, as in [1]. A feedback policy is able to respond to disturbances online; recognition of this fact helps reduce conservatism when the controller is synthesized subject to robust constraints and performance objectives, involving 'worst-case disturbances'. Second, we extend the minimax formulation of [1] to consider a wider variety of costs, constraints, and sets of allowable disturbances. We make use of recent developments in convex formulations of polytopic containment problems [8], to derive more computationally efficient formulations of the minimax robust synthesis problem, for polytopically constrained disturbances.

Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA. E-mail: umnbrgr@mit.edu.

A. Related work

The foundation for the specific approach to data-enabled (a.k.a data-driven) control pursued in this paper can be traced back to the work of Willems et al. [9], which proved that all (input-output) trajectories of a linear dynamical system lie in an affine subspace formed by a finite subset of those trajectories, subject to persistence of excitation conditions; cf. Theorem 4. Shortly thereafter, this result was exploited for data-driven simulation and control, cf. e.g. [10].

Recently, there has been a resurgence of interest in the application of [9] to data-driven control. The so-called '**D**ata-**e**nabl**e**d **P**redictive **C**ontrol' approach, a.k.a. 'DeePC', was introduced in [11]. DeePC builds upon [10], with a greater focus on constrained optimal control. A distributionally robust version of DeePC was presented in [12], which gave a principled interpretation to the regularization heuristics proposed in [11]. Further robustness results were presented in [13], [14], while questions concerning the necessity of persistence of excitation were investigated in [15]. Of greatest relevance to the present paper is [1], which extends DeePC to handle measurable disturbances; cf. §II-B for a detailed problem formulation. We build on this work by deriving convex formulations of the search for closed-loop policies, rather than open-loop input sequences. To do so, we make use of a parametrization of closed-loop policies known as *disturbance feedback*, cf. e.g., [16], [17], which has roots dating back to the 1970s [18].

II. PRELIMINARIES

A. Notation and terminology

The transpose of a vector a is denoted a'. For a sequence $\{x_k\}_{k=1}^N$, with $x_k \in \mathbb{R}^n$, let $x_{a:b} := [x'_a, \dots, x_b]'$, i.e., the column vector obtained by stacking elements of the sequence, from a to b, $b \geq a, \in \mathbb{Z}$. When there is no risk of ambiguity, we drop the subscript when the vectorized representation includes all elements of the sequence, e.g., for $\{x_k\}_{k=1}^N$, $x_{1:N}$ is equivalent to x. We will often refer to a sequence $\{x_k\}_{k=1}^N$ by its 'vectorized' notation $x_{1:N}$, for brevity. For a sequence $x_{1:N}$ let $\mathcal{H}_L(x)$ denote the block Hankel matrix

$$\mathcal{H}_L(x) = \begin{bmatrix} x_1 & x_2 & \dots & x_{N-L+1} \\ x_2 & x_3 & \dots & x_{N-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & \dots & x_N \end{bmatrix}.$$

A signal $x_{1:N}$, $x_k \in \mathbb{R}^n$, is said to be persistently exciting of order L if $\text{rank}\,(\mathcal{H}_L(x)) = nL$, i.e., full row rank. For $x \in \mathbb{R}^n$, let $\|x\|_p := (\sum_{i=1}^n |x(i)|^p)^{\frac{1}{p}}$, where $x(i)$ is the i-th element of x. The n-dimensional p-norm unit ball is denoted $\mathbb{B}_p^n := \{x \in \mathbb{R}^n \mid \|x\|_p \le 1\}$. $\|x\|_Q^2$ is shorthand for $x'Qx$.

B. Problem statement

We are interested in controlling the following system,

$$x_{t+1} = f(x_t, u_t, w_t), \quad y_t = g(x_t, u_t, w_t), \tag{1}$$

where $x_t \in \mathbb{R}^{n_x}$, $u_t \in \mathbb{R}^{n_u}$, $w_t \in \mathbb{R}^{n_w}$, $y_t \in \mathbb{R}^{n_y}$ denote the internal state, input, disturbance, and output respectively, at time t. We make the following crucial assumption:

Assumption 1. *The dynamical system (1) can be approximated with sufficient accuracy,* for the purpose of control*, by a linear time invariant (LTI) model, possibly of higher order.*

While this assumption excludes a number of important and interesting nonlinear systems, approximation of nonlinear dynamics with linear models is not uncommon in modeling and control, particularly in the context of Koopman operator theory, cf. e.g. [19]. Furthermore, Assumption 1 clearly encompasses the special case where (1) is, in fact, linear.

Our goal is to track the (known) reference signal $r_{0:T}$ over the time horizon $t = 0, \dots, T$. Tracking error is quantified by a cost function, $c_r(y_{0:T} - r_{0:T})$. Possible cost functions include, e.g., $c_r(\cdot) = Q\|\cdot\|_p$, $p = 1, 2, \infty$, where $Q \in \mathbb{R}_+$ is a nonnegative user-specified weight. Control inputs may also be penalized via the cost function $c_u(u_{0:T})$. As with the tracking cost, popular cost functions include, e.g., $c_u(\cdot) = R\|\cdot\|_p$, $p = 1, 2, \infty$, where $R \in \mathbb{R}_+$ is a nonnegative user-specified weight. The total cost to be minimized is $c_r(y_{0:T} - r_{0:T}) + c_u(u_{0:T})$.

For $0 \le t_1,\, t_2 \le T$, control inputs $u_{t_1:t_2}$ are restricted to the convex set $\mathcal{U}_{t_1:t_2}$. For brevity, the subscript ${}_{t_1:t_2}$ will be dropped from $\mathcal{U}_{t_1:t_2}$ when it is clear from context. Popular choices of $\mathcal{U}$ include, e.g., amplitude bounds: $u_t^{\text{min}} \le u_t \le u_t^{\text{max}}$ for $t \in 0 : T$. Similarly, system outputs $y_{t_1:t_2}$ are constrained to the convex set $\mathcal{Y}_{t_1:t_2}$.

The true system (1) is unknown; i.e., we do not know f or g. We do, however, assume access to input-output data from the system, specifically, a trajectory $\mathcal{D} := \{u_t^{\text{d}}, w_t^{\text{d}}, y_t^{\text{d}}\}_{t=1}^N$ satisfying (1). Notice that we have assumed that the disturbances are observable.

Assumption 2 (Observable disturbances, [1])**.** *All past disturbances are assumed to be known. Specifically, during control of the system, at time t all past disturbances $w_{0:t-1}$ and disturbances $w_{1:N}^{\text{d}}$ in $\mathcal{D}$ are known.*

We wish to emphasize that this assumption is lifted directly from [1], upon which we build. Assumption 2 certainly does not hold in all applications; nevertheless, there are some settings in which disturbances can indeed be measured, e.g. control of low-frequency oscillations in power systems [1], or e.g. control of wind turbines equipped with wind speed sensors, cf. also [20].

We will further assume that the disturbances are bounded.

Assumption 3 (Bounded disturbances)**.** *For any subset $t_1, \dots, t_2$ of the duration under which the system is controlled, i.e, $0 \le t_1, t_2 \le T$, the disturbances $w_{t_1:t_2}$ are confined to a known bounded, convex set $\mathcal{W}_{t_1:t_2}$.*

As with $\mathcal{U}$, we will often drop the subscript ${}_{t_1:t_2}$ from $\mathcal{W}_{t_1:t_2}$ when it is clear from context.

In addition to $\mathcal{D}$, we will also assume that, when we initiate control of the system at time $t = 0$, we have observed the past T_i inputs, disturbances, and outputs $\mathcal{I} := \{u_t, w_t, y_t\}_{t=-T_i}^{-1}$ from the system. As will be clarified in §II-C, cf. Remark 6, these observations effectively specify the 'initial conditions' of the system.

We may now specify the control problem addressed in this paper: minimization of the trajectory tracking cost, for worst-case disturbances, i.e.,

$$\min_{u_{0:T} \in \mathcal{U}, y_{0:T}} \max_{w_{0:T} \in \mathcal{W}_{0:T}} c_r(y_{0:T} - r_{0:T}) + c_u(u_{0:T}), \tag{2}$$

such that $\{u_t, w_t, y_t\}_{t=-T_i}^T$ satisfy (1), i.e., constitute a valid trajectory of the system. Notice that $\{u_t, w_t, y_t\}_{t=-T_i}^T$ includes the 'initial conditions' in $\mathcal{I}$.

C. Data-enabled approach to control

In the absence of knowledge of the true system parameters, a common approach to solving (2) would first involve 'learning' an approximate system model from the supplied data $\mathcal{D}$ via system identification, and then proceeding with model-based control design.

In this paper, we adopt the so-called '**D**ata-**e**nabl**e**d **P**redictive **C**ontrol' approach, a.k.a. 'DeePC'. In DeePC, one assumes that the system (1) can be approximated, for the purpose of feedback control, sufficiently accurately by a linear time-invariant (LTI) model. Rather than 'learn' an explicit linear model via system identification, in DeePC one models future trajectories of the system as an (affine) function of an observed, past trajectory, $\mathcal{D}$, under the assumption that these trajectories - both past and future - are consistent with the same underlying LTI system. In particular, one has the following key result:

Theorem 4 (Data-enabled simulation [9])**.** *Consider an LTI system with the following minimal realization*

$$s_{t+1} = As_t + Bu_t, \quad y_t = Cs_t + Du_t, \tag{3}$$

with state $s_t \in \mathbb{R}^{n_s}$. Suppose $\{\tilde{u}_t, \tilde{y}_t\}_{t=1}^N$ is a trajectory of (3). Let $\tilde{u}$ be persistently exciting of order $L + n_s$. Then $\{\bar{u}_t, \bar{y}_t\}_{t=1}^L$ is a trajectory of (3) if and only if there exists

$g \in \mathbb{R}^{N-L-n_s+1}$ *such that*

$$\begin{bmatrix} \mathcal{H}_T(\tilde{u}) \\ \mathcal{H}_T(\tilde{y}) \end{bmatrix} g = \begin{bmatrix} \bar{u} \\ \bar{y} \end{bmatrix}. \quad (4)$$

Persistence of excitation of order $L + n_s$ is required to have enough degrees of freedom to encode the length-L input $\bar{u}_{1:L}$, and the initial condition $\bar{s}_1 \in \mathbb{R}^{n_s}$.

To handle disturbances, w_t, as in (1), one can consider

$$s_{t+1} = As_t + Bu_t + B_w w_t, \quad y_t = Cs_t + Du_t + D_w w_t, \quad (5)$$

with state $s \in \mathbb{R}^{n_s}$, and $(A, [B \; B_w])$ controllable. Observe that, compared to (3), we have introduced the disturbance $w_t \in \mathbb{R}^{n_w}$. Notice also that we allow the state $s \in \mathbb{R}^{n_s}$ of the linear system (5) to be of different dimension to the state $x \in \mathbb{R}^{n_x}$ of the system (1). To approximate (a possibly nonlinear) system (1) with the linear system (5), it may be necessary to use a state of higher dimension in the latter, i.e., $n_s > n_x$.

By treating the disturbance w as an additional 'input' (i.e., consider an augmented input given by the concatenation of u_t and w_t), one can apply Theorem 4 to (5). In particular, if one assumes that the past data $\mathcal{D} = \{u^{\rm d}, w^{\rm d}, y^{\rm d}\}$ represents a valid trajectory for (5), with $\{u_t^{\rm d}, w_t^{\rm d}\}_{t=1}^N$ persistently exciting of order $L + T_i$, then the sequence $\{u_t, w_t, y_t\}_{t=-T_i}^{L}$ is a trajectory of (5) if and only if there exists $g \in \mathbb{R}^{N-L-T_i+1}$ such that

$$\begin{bmatrix} U_p \\ W_p \\ Y_p \\ U_f \\ W_f \\ Y_f \end{bmatrix} g = \begin{bmatrix} u_{-T_i:-1} \\ w_{-T_i:-1} \\ y_{-T_i:-1} \\ u_{0:L-1} \\ w_{0:L-1} \\ y_{0:L-1} \end{bmatrix}, \quad (6)$$

where

$$\begin{bmatrix} U_p \\ U_f \end{bmatrix} := \mathcal{H}_{T_i+L}(u^{\rm d}), \begin{bmatrix} W_p \\ W_f \end{bmatrix} := \mathcal{H}_{T_i+L}(w^{\rm d}), \quad (7)$$
$$\begin{bmatrix} Y_p \\ Y_f \end{bmatrix} := \mathcal{H}_{T_i+L}(y^{\rm d}).$$

In (7), U_p, W_p, Y_p each have T_i block-rows, and U_f, W_f, Y_f each have L block-rows. The relation in (6) follows from (4) in Theorem 4 with appropriate substitutions, e.g., $\tilde{u}_{1:N} = \{u_t^{\rm d}, w_t^{\rm d}\}_{t=1}^N$.

Remark 5 (Necessary conditions for persistence of excitation)**.** *The parametrization of trajectories of (5) in (7) is only valid when $v^{\rm d} := \{u_t^{\rm d}, w_t^{\rm d}\}_{t=1}^N$ persistently exciting of order $X = T_i + L + n_s$. As $\mathcal{H}_X(v^{\rm d})$ has $X(n_u + n_w)$ rows and $N - X + 1$ columns, it can only be full (row) rank when $N - X + 1 \geq X(n_u + n_w)$. This implies $N \geq (T_i + L + n_s)(n_u + n_w)$.*

In light of Remark 5, it is clear that the length of the observed trajectory $\mathcal{D}$ must satisfy $N \geq (T_i + L + n_s)(n_u + n_w)$ to ensure validity of (6).

Remark 6 (Initial conditions)**.** *For the linear model (3), one has the (well-known) relation: $[y_1', y_2', \ldots, y_\ell']' =$*

$$\underbrace{\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{\ell-1} \end{bmatrix}}_{\mathcal{O}} s_1 + \underbrace{\begin{bmatrix} D & 0 & \cdots \\ CB & D & \cdots \\ \vdots & \ddots & \\ CA^{\ell-2}B & \cdots & D \end{bmatrix}}_{\mathcal{G}} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_\ell \end{bmatrix}. \quad (8)$$

The smallest $\ell \in \mathbb{N}$ such that $\mathcal{O}$ has rank n_s is known as the 'lag' of the system. When $\mathcal{O}$ is full column rank one can solve for s_1 given $u_{1:\ell}$, $y_{1:\ell}$ (and $\mathcal{O}$, $\mathcal{G}$).

Assuming a linear model (5) with lag ℓ, Remark 6 implies that one should choose $T_i \geq \ell$ such that $\mathcal{I} = \{u_{-T_i:-1}, w_{-T_i:-1}, y_{-T_i:-1}\}$ uniquely specify the initial conditions for the subsequent trajectory $\{u_{0:L}, w_{0:L}, y_{0:L}\}$.

Given $\mathcal{D}$ and $\mathcal{I}$, (6) constitutes affine constraints on $\{u_{0:L}, w_{0:L}, y_{0:L}\}$. Assuming a linear model (5) of the unknown system (1), one can solve the minimax reference tracking problem in a receding-horizon fashion, by searching over $\{u_{0:L}, w_{0:L}, y_{0:L}\}$ subject to this affine constraint, using convex programing. This is the approach taken in [1], which constitutes an open-loop predictive control strategy, as one searches for an input sequence $u_{0:L}$. In the following section, we propose a closed-loop alternative that searches over sequences of feedback policies, rather than sequences of inputs.

III. Closed-loop DeePC

It is widely appreciated that if one wishes to account for disturbances in a constrained predictive control problem, better performance can be obtained by optimizing over sequences of feedback policies rather than sequences of inputs [21]. In essence, designing a sequence of (future) inputs does not take into account the fact that additional information - namely, estimates of (now) past disturbances - will be available at the (future) time when those inputs are applied to the system.

A. Policy parametrization

We adopt a closed-loop, data-enabled predictive control approach to solving problem (2). Specifically, at time t we wish to determine a sequence of $L + 1$ policies $\pi_{0:L}^t = \{\pi_k^t\}_{k=0}^L$ to minimize cost over the horizon $t, \ldots, t + L$, i.e.,

$$\min_{\pi_{0:L}^t \in \mathcal{U}, y_{0:L}^t} \; \max_{w_{t:t+L} \in \mathcal{W}_{t:t+L}} \; c_r(y_{0:L}^t - r_{t:t+L}) + c_u(\pi_{0:L}^t) \quad (9)$$

such that $\{u_{t-T_i:t-1}, \pi_{0:L}^t\}$, $\{w_{t-T_i:t-1}, w_{t:t+L}\}$, and $\{y_{t-T_i:t-1}, y_{t:t+L}^t\}$ satisfy (5), i.e., constitute a valid trajectory of the approximate linear model. We make the following clarifying remarks:

i. This constitutes a standard receding-horizon (of length $L+1$) approach to solving constrained optimal control problems, such as (2).
ii. In the spirit of DeePC, we approximate the true unknown system (1), with the linear model (5). The approximation

is implicit, in that we use (6) to make predictions, rather than fitting an explicit linear model.

iii. Trajectories with the subscript ${}_{t-T_i:t-1}$ denote past T_i observed values of the system trajectory, and serve as initial conditions for the predicted trajectories at time t.

iv. We use the superscript ^{t} to distinguish the predicted future trajectories at time t (e.g. the predicted output $y^t_{0:L}$), from the actual system trajectories, (e.g. $y_{t:t+L}$).

v. Notice that policies $\pi^t_{0:L}$ have replaced inputs $u_{t:t+L}$ as the quantities for which we search.

We will parametrize the policies $\pi^t_{0:L}$ as follows:

$$\pi^t_0 = \nu^t_0, \quad \pi^t_k(w_{t:t+k-1}) = \nu^t_k + \sum_{j=0}^{k-1} K^t_{k,j} w_{t+j}, \ \forall k \in 1:L. \tag{10}$$

This policy is simpler to interpret when expressed as:

$$\pi^t_{0:L} = \begin{bmatrix} \pi^{t\prime}_0 & \pi^{t\prime}_1 & \pi^{t\prime}_2 & \dots & \pi^{t\prime}_L \end{bmatrix}' =$$

$$\underbrace{\begin{bmatrix} \nu^t_0 \\ \nu^t_1 \\ \nu^t_2 \\ \vdots \\ \nu^t_L \end{bmatrix}}_{\nu^t} + \underbrace{\begin{bmatrix} 0 & \cdots & & \\ K_{1,0} & 0 & \cdots & \\ K_{2,0} & K_{2,1} & 0 & \\ \vdots & \vdots & & \\ K_{L,0} & \dots & K_{L,L-1} \end{bmatrix}}_{=:\mathcal{K}^t} \begin{bmatrix} w_t \\ w_{t+1} \\ w_{t+2} \\ \vdots \\ w_{t+L-1} \end{bmatrix}.$$

Here, ν^t denotes a nominal input sequence, while $\mathcal{K}^t$ denotes a matrix of disturbance feedback gains. Notice that the policy π^t_k, which is intended to be applied at time $t+k$, depends only on disturbances w_{t+k-1} and earlier, which are observable at time $t+k$ by Assumption 2. As such, the policy is causal and can be implemented online.

Remark 7. *In (10), our notation makes it clear that the policy π^t_k depends explicitly on the disturbances $w_{t:t+k-1}$. A natural question may be: should not the policy depend on all past disturbances? This dependence is present, but implicit. Specifically, as shall be made clear in §III-B, the decision variables $\nu^t, \mathcal{K}^t$ are designed to satisfy (6), which introduces dependence on the past T_i disturbances, $w_{t-T_i:t-1}$, as well as w^{d}.*

B. Data-enabled prediction of system output

To solve the receding horizon problem (2), we must predict the output of the system over the horizon $t : t+L$. By (6) the (predicted) output of the linear model (5) is given by $y^t_{0:L} = Y_f g$ (final block row of (6)) where g must satisfy

$$\underbrace{\begin{bmatrix} U_p \\ W_p \\ Y_p \\ U_f \\ W_f \end{bmatrix}}_{=:D} g = \underbrace{\begin{bmatrix} u_{t-T_i:t-1} \\ w_{t-T_i:t-1} \\ y_{t-T_i:t-1} \\ \pi^t_{0:L} \\ w_{t:t+L} \end{bmatrix}}_{=:d} = \begin{bmatrix} \begin{bmatrix} \\ \xi^t \\ \\ \end{bmatrix} \\ \pi^t_{0:L} \\ w_{t:t+L} \end{bmatrix}. \tag{11}$$

Here D is formed from the given data $\mathcal{D}$, and is, therefore, known. The quantity $\xi^t = [u'_{t-T_i:t-1}, w'_{t-T_i:t-1}, y'_{t-T_i:t-1}]'$ denotes previously observed inputs, disturbances, and outputs, which effectively specify the initial conditions of the model, cf. Remark 6. $\pi^t_{0:L}$ denotes the sequence of policies for which we will search. $w_{t:t+L}$ denotes the sequence of future disturbances, which, as of time t, are currently unknown, but bounded by Assumption 3.

A parametrization of solutions g for (11) is given by $g = D^\dagger d + D^\perp z$, where $D^\dagger$ denotes the pseudeo-inverse of D such that $D^\dagger d$ gives a particular solution to (11), and $D^\perp = I - D^\dagger D$ such that $D^\perp z$ parametrizes the nullspace of D for arbitrary z. By partitioning $D^\dagger = [D_\xi \ D_\pi \ D_w]$ such that $D^\dagger d = D_\xi \xi + D_\pi \pi^t + D_w w_{t:t+L}$, we have the following parametrization of the predicted output

$$y^t = \underbrace{Y_f D_\xi}_{=:M_i} \xi + \underbrace{Y_f D_\pi}_{=:M_\pi} \pi^t + \underbrace{Y_f D_w}_{=:M_w} w_{t:t+L} + D^\perp z, \tag{12}$$

which is affine in both the decision variables $\{\pi^t, z\}$ and the unknown disturbances $w_{t:t+L}$.

C. Policy synthesis

For clarity of exposition, we shall now derive a convex synthesis procedure for feedback policies of the form (10), for specific choices costs c_r, c_u and constraints $\mathcal{U}, \mathcal{Y}$. This will allow us to introduce the key ideas and machinery, without the burden of considering the problem in full generality. Synthesis details for alternative choices of costs and constraints are provided in §III-D.

Specifically, we choose $c_r(\cdot) = \|\cdot\|_\infty$, to penalize maximum deviation from the reference trajectory r. We shall ignore c_u, as it is straightforward to include any of the convex costs on the input listed in §II-B. We will also assume H-polytopic representations for $\mathcal{U}$ and $\mathcal{W}$, i.e., $\mathcal{U} = \{u \in \mathbb{R}^{(L+1)n_u} \mid H_u u \le h_u\}$, $\mathcal{W} = \{w \in \mathbb{R}^{(L+1)n_w} \mid H_w w \le h_w\}$. For clarity, we ignore constraints ($\mathcal{Y}$) on the outputs; cf. §III-D.1 for details. Given these choices, the predictive control problem introduced in (9) reduces to

$$\min_{y^t,\nu,\mathcal{K},z} \ \max_{w\in\mathcal{W}} \ \|\underbrace{M_i\xi + M_\pi(\nu + \mathcal{K}w) + M_w w + D^\perp z}_{y^t} - r\|_\infty$$
$$\text{s.t. } H_u(\nu + \mathcal{K}w) \le h_u \ \forall \ w \in \mathcal{W} = \{w \mid H_w w \le h_w\}, \tag{13}$$

where $\pi = \nu + \mathcal{K}w$ is the policy parametrization from (10), and $y^t = M_i\xi + M_\pi \pi + M_w w + D^\perp z$ is the parametrization of the predicted output from (12). We are now in a position to present the main contribution of this section: a convex program that minimizes an upper bound on (13).

Theorem 8. *Minimization of an upper bound for (13) can be formulated as the following (convex) linear program:*

$$\min_{\zeta\in\mathbb{R},\nu,\mathcal{K},z,\Lambda^r,\beta^r,\Lambda^u,\beta^u} \quad \zeta \tag{14a}$$
$$\text{s.t. } \Lambda^r H_w = H_e(M_w + M_\pi \mathcal{K}), \ \Lambda^r \ge 0 \tag{14b}$$
$$\Lambda^r h_w \le \zeta h_e + H_e \beta^r, \tag{14c}$$
$$-\beta^r = M_i\xi + M_\pi \nu + D^\perp z - r_{t:t+L}, \tag{14d}$$
$$\Lambda^u H_w = H_u \mathcal{K}, \ \Lambda^u h_w \le h_u - H_u \nu, \tag{14e}$$

where $H_e = [I_n, -I_n]'$ and $h_e = \mathbf{1}_{2n}$, with $n = (L+1)n_y$, such that $\mathbb{B}^n_\infty = \{x \in \mathbb{R}^n \mid \mid H_e x \le h_e\}$. Dimensionality of all decision variables can be inferred from context.

To prove Theorem 8 we will make use of two recent results on polytopic containment.

Lemma 9 (Polytope containment [8])**.** *Given two sets* $\mathbb{Q}_i = \bar{x}_i + G_i\mathbb{H}_i \subseteq \mathbb{R}^n$, $\mathbb{H}_i = \{x \in \mathbb{R}^n \mid H_i x \leq h_i\}$, $i = 1, 2$, *a sufficient condition for* $\mathbb{Q}_1 \subseteq \mathbb{Q}_2$ *is existence of* Γ, Λ, *and* β *such that*

$$G_1 = G_2\Gamma,\ \Lambda H_1 = H_2\Gamma, \tag{15a}$$

$$\bar{x}_2 - \bar{x}_1 = G_2\beta,\ \Lambda h_1 \leq h_2 + H_2\beta,\ \Lambda \geq 0. \tag{15b}$$

Notice that Lemma 9 only provides sufficient conditions for containment; this is a possible source of conservatism in our formulation, and the reason why (14) is an upper bound.

Lemma 10 (Minimax via containment)**.** *Let* $\mathbb{H}$ *denote a bounded H-polytope, and* $\mathbb{Q} = \{x \in \mathbb{R}^n \mid x = \bar{x} + G\omega,\ \omega \in \mathbb{H}\}$. *Then*

$$\max_{\omega\in\mathbb{H}} \|\bar{x} + G\omega\|_p = \min_{\zeta\in\mathbb{R}} \zeta, \quad \text{s.t. } \mathbb{Q} \subseteq \zeta\mathbb{B}_p^n \tag{16}$$

Proof. By considering the epigraph formulation, the LHS of (16) is equivalent to $\min_{\zeta,\omega}\ \zeta$, s.t. $\zeta \geq \|\bar{x} + G\omega\|_p\ \forall\ \omega \in \mathbb{H}$. Then $\zeta \geq \|\bar{x} + G\omega\|_p\ \forall\ \omega \in \mathbb{H} \iff \mathbb{Q} \subseteq \zeta\mathbb{B}_p^n$ completes the proof. $\square$

Turning now to (14), conditions (14b)–(14d) encode the minimax cost function, using Lemma 10. Specifically, by making use of (12), the minimax problem can be expressed as minimization of the slack variable ζ subject to a polytopic containment constraint, as in (16), with $\bar{x} = M_i\xi + M_\pi\nu + D^\perp z - r_{t:t+L}$, $G = M_w + M_\pi\mathcal{K}$, $\omega = w$, and $\mathbb{H} = \mathcal{W}$. A sufficient condition for the containment in (16) is given by Lemma 9, and (14b)–(14d) follow directly from (15) with $\{\bar{x}_1, G_1\} = \{\bar{x}, G\}$ (as above), $\{\bar{x}_2, G_2\} = \{0, I\}$, $\mathbb{H}_1 = \mathbb{H} = \mathcal{W}$, $\mathbb{H}_2 = \mathbb{B}_\infty^n$, and $p = \infty$. Notice that $\zeta\mathbb{B}_\infty^n = \{x \in \mathbb{R}^n| \mid H_e x \leq \zeta h_e\}$, and so the conditions remain affine in the decision variables. We reiterate that Lemma 9 provides only sufficient conditions for containment; this possible conservatism means that (14) constitutes an upper bound for the problem. Condition (14e) encodes the input constraints, and follows by application of Lemma 9 to the containment problem: $\nu + M_\pi\mathcal{K}w \in \mathcal{U}$ for all $w \in \mathcal{W}$. $\square$

We close this subsection with some brief remarks on the difference between the open-loop and closed-loop formulations. First, observe that the open-loop problem is recovered as a special case of (14), with $\mathcal{K} \equiv 0$. Consequently, the closed-loop formulation can only increase the size of the feasible set for the synthesis program, so the solution will always be at least as good as the open-loop formulation. More precisely, consider (14b). Here it is clear that $\mathcal{K}$ increases the size of the feasible set of Λ that must satisfy this equality constraint, thereby reducing conservatism (relative to the open-loop case in which $\mathcal{K} \equiv 0$).

Second, the increase in computational complexity associated with the proposed closed-loop formulation is modest: specifically, the open-loop and closed-loop formulations result in the same class of optimization problem (e.g. an LP in Theorem 8). The additional complexity of closed-loop synthesis comes from introduction the additional decision variable(s) $\mathcal{K}$, the dimension of which can be controlled by imposing structure, e.g. sparsity or repeated gains $K_{i,j}$ in (10). Furthermore, depending on the choice of cost function, the proposed closed-loop implementation can actually be more computationally tractable than [1], as, e.g. in the case of quadratic costs, it avoids introduction of a constraint for each unknown future disturbance, cf. §III-D.2 for details.

Remark 11. *When the true dynamical system (1) is indeed nonlinear, or when the observations are corrupted by unobserved disturbances, it is necessary to introduce a regularization term to the objective (14a), cf. [12]. Specifically, one should regularize the solution g to (6); cf. §III-E for further details and discussion.*

D. Alternate problem formulations: costs and constraints

In §III-C, we derived a synthesis procedure based on convex programing for a specific problem instance, namely, the infinity-norm tracking cost $c_r(\cdot) = \|\cdot\|_\infty$. In this section, we briefly sketch the problem formulation for a number of alternative cost functions and constraint sets. For brevity of notation, we define $N_y = (L+1)n_y$ such that $y \in \mathbb{R}^{N_y}$. Similarly for N_u and N_w. Furthermore, to approximate the minimax objective it is convenient to introduce

$$f := M_i\xi + M_\pi\nu + D^\perp z - r, \quad F := (M_w + M_\pi\mathcal{K}), \tag{17}$$

so that the tracking error can be written as an affine function of the disturbances, i.e., $y - r = Fw + f$, cf. (12). Note also, that both f and F are affine functions of the decision variables $\nu, z, \mathcal{K}$.

1) Constraints on outputs: It is straightforward to enforce that the output y remain in some polytopic set $\mathcal{Y}$ for all $w \in \mathcal{W}$, when $\mathcal{W}$ is also polytopic. Specifically, let $\mathcal{Y} = \mathbb{H}_y := \{y \in \mathbb{R}^{N_y} \mid H_y y \leq h_h\}$ be a (bounded) polytope representing admissible values for the output. Given the affine dependence of the output y on policies π and disturbances w, cf. (12), we have: $y \in \mathcal{Y} \iff$

$$(M_i\xi + M_\pi\nu + (M_w + M_\pi\mathcal{K})w + D^\perp z) \in \mathbb{H}_y,\ \forall\ w \in \mathcal{W}.$$

A sufficient condition for this containment is given by Lemma 9, which results in affine constraints on the decision variables $\pi = \{\nu, K\}$ and z.

2) Quadratic costs, polytopic $\mathcal{W}$: The approach proposed in [1], upon which the present paper builds, assumed quadratic costs - i.e, $c_r = \|y - r\|_Q^2$ - and bounded disturbances: $w^{\min} \leq w_t \leq w^{\max}$ for all t. It is well-known that the maxima of a convex function over a bounded polytopic set occur at a (subset) of the vertices of the polytopic set. As such, a tight upper bound for $\max_{w\in w}\ \|Fw + f\|_Q^2$ - note the use of (17) - can be obtained by introducing a slack variable $\zeta \in \mathbb{R}$, and constraints $\zeta \geq \|Fw + f\|_Q^2$ for $i = 1 : 2^{N_w}$, where $\hat{w}^i$ denotes the i-th vertex of $\mathcal{W}$. As $Fw + f$ is affine in the decision variables, these constraints define a convex set; however, the number of constraints grows exponentially in N_w. A 'downsampling' procedure to reduce the dimensionality of $w \in \mathbb{R}^{N_w}$ was proposed in [1].

This construction allows the minimax objective to be minimized to global optimality, at the expense of exponentially (in N_w) many constraints. As an alternative, one can formulate a more computationally tractable upper bound, at the expense of conservatism, using the S-procedure. As above, we can form an upper bound by introducing a slack variable ζ such that $\zeta \geq \|Fw + f\|_2^2$ holds for all $w \in \mathcal{W}$. By the Schur complement, this condition is equivalent to

$$\mathcal{F} + \mathcal{L}\Delta\mathcal{R} + \mathcal{R}'\Delta\mathcal{L}' \succeq 0, \quad \forall\, \Delta \in \boldsymbol{\Delta}, \tag{18}$$

where

$$\mathcal{F} = \begin{bmatrix} \zeta & f' \\ f & I \end{bmatrix},\ \mathcal{L} = \begin{bmatrix} 0 \\ F \end{bmatrix},\ \mathcal{R} = \begin{bmatrix} I & 0 \end{bmatrix},\ \Delta = \mathrm{diag}(w),$$

and $\boldsymbol{\Delta} = \{\mathrm{diag}(w) \in \mathbb{R}^{N_w \times N_w} \mid |w(i)| \leq 1,\ i = 1 : N_w\}$ encodes the set of feasible disturbances $\mathcal{W} = \mathbb{B}_\infty^{N_w}$. By (a straightforward modification of) [22, Lemma 3.1] a sufficient condition for (18) is the existence of $\mathcal{T} = \mathrm{diag}(\tau)$ with multiplier $\tau \in \mathbb{R}^{N_w}$ such that

$$\begin{bmatrix} \mathcal{F} - \mathcal{R}'\mathcal{T}\mathcal{R} & \mathcal{L}' \\ \mathcal{L} & \mathcal{T} \end{bmatrix} \succeq 0. \tag{19}$$

Notice that (19) is linear in the slack variable ζ, multiplier τ, and f, F, which are in turn affine in the decision variables ν, K, z. Therefore, minimization of the upper bound ζ subject to (19) is a semidefinite program (SDP).

3) Quadratic costs, disturbances of bounded energy: In some applications, it may be more appropriate to consider disturbances bounded by some elliptical set, e.g., to encode 'finite energy' constraints on the disturbance. Specifically, consider $\mathcal{W} = \{w \in \mathbb{R}^{N_w} \mid w'Pw + 2p'w + q \leq 0\}$, where $P \in \mathbb{S}_+^{N_w}$, $p \in \mathbb{R}^{N_w}$, and $q \in \mathbb{R}$ are known. By denoting $f := M_i\xi + M_\pi\nu + D^\perp z - r$ and $F := (M_w + M_\pi\mathcal{K})$, we can express the tracking error as $y - r = Fw + f$, cf. (12). A quadratic cost on this error is given by $\|Fw + f\|_Q^2$. Minimizing this quadratic cost for all $w \in \mathcal{W}$ is then equivalent to minimizing an upper bound $\zeta \geq \|Fw + f\|_Q^2$ that holds for all $w \in \mathcal{W}$. To this end, by application of the S-procedure, we have: $\zeta \geq \|Fw + f\|_Q^2 \impliedby w \in \mathcal{W}$ if and only if there exists $\tau \in \mathbb{R}_+$ such that

$$\tau \begin{bmatrix} P & p \\ p' & q \end{bmatrix} - \begin{bmatrix} F'QF & F'Qf \\ f'QF & f'Qf - s \end{bmatrix} \succeq 0. \tag{20}$$

This matrix inequality is nonlinear (quadratic) in the decision variables implicit in F and f; however, by application of the Schur complement, (20) is equivalent to

$$\begin{bmatrix} \mathcal{P} & \Theta \\ \Theta' & I \end{bmatrix} \succeq 0,\ \underbrace{\begin{bmatrix} \tau P & \tau p \\ \tau p' & \tau q + \zeta \end{bmatrix}}_{\mathcal{P}},\ \underbrace{\begin{bmatrix} Q^{\frac{1}{2}}F & Q^{\frac{1}{2}}f \\ 0 & 0 \end{bmatrix}}_{\Theta}. \tag{21}$$

Condition (21) is an LMI, and so $\min_{\zeta,\tau,F,f}\ \zeta$, s.t. (21) is a convex program (SDP).

4) Nominal disturbances: Finally, a popular choice of objective in robust receding horizon control, is to assume that the disturbances take on some nominal value, denoted $\bar{w}$, cf. e.g. [16, §7.1]. Here, $\bar{w}$ could represent the expected value of w if one can assume some distribution over disturbances; alternatively, it could simply denote some likely value of the disturbance, e.g., zero. With the parametrization of the output in (12), we have a tracking cost

$$c_r(M_i\xi + M_\pi\nu + (M_w + M_\pi\mathcal{K})\bar{w} + D^\perp z - r),$$

which is convex in the decision variables; no polytopic containment problem need be solved. Constraints $u \in \mathcal{U}$ and $y \in \mathcal{Y}$ could still be enforced robustly (i.e., for all $w \in \mathcal{W}$), however, the cost function is 'optimistic' in that we optimize for nominal, rather than worst-case, disturbances.

E. Regularization

In Remark 11 we mentioned that careful regularization is necessary to achieve good performance in the DeePC approach. Here we elaborate on that point. Given a nominal model, any optimal control synthesis procedure will exploit the properties of this model to obtain the lowest possible cost. However, this is merely the cost of the synthesis procedure given the nominal model, which may be different to the cost of deploying the controller on the true system, due to inaccuracies in the nominal model. Ensuring satisfactory performance on the true system despite errors in the modeling process is the premise of robust control.

In DeePC, the affine subspace (6) plays the role of (an implicit) model, in that it is used to predict system behavior. DeePC then optimizes *jointly* over the 'controller' (e.g. a sequence of inputs or feedback policies) and the resulting system response. As this optimization occurs jointly, DeePC effectively searches for the 'best case' (as measured by the cost function) system response that is consistent with the observed data. When the true system - that generated the data $\mathcal{D}$ - is not LTI (as assumed), it is essential to regularize the predicted response, to constrain the extent to which DeePC can exploit the inaccuracies in the 'model' (6).

One such regularization strategy is to introduce a penalty term such as $\lambda\|g\|_p$ to the control objective. Here, g (an implicit decision variable, cf. (12)) denotes the solution of (11), $p \in \{1, 2\}$ specifies the norm, and λ is the user-specified regularization weight. A principled justification for this regularization strategy in the context of 'distributionally robust' optimization is given in [12]; a similar strategy is used in [13]. Tuning λ appropriately is essential for good performance; however, in practice, this tuning must be conducted by trial and error (though some qualitative guidelines are provided in [13]). More principled methods for tuning regularization parameters in data-enabled control is an important and interesting direction for future research.

IV. Numerical illustration

In this section we illustrate the advantages of the proposed closed-loop DeePC approach compared to existing open-loop DeePC. The true system to be controlled is of the form (1)

$(n_x = 4,\ n_u = 1,\ n_w = 1,\ n_y = 1)$ with

$$f(x,u,w) = \begin{bmatrix} \theta' x + c \times \phi(x(1)) + u + 0.2w \\ \Theta x \end{bmatrix},$$
$$\phi(s) = 0.2s + 0.1s^2 + 0.6s^3$$
$$y = x(1) + 0.05x(1)^3 + 0.05w. \tag{22}$$

For values of the parameters $\theta \in \mathbb{R}^4$, $\Theta \in \mathbb{R}^{3\times 4}$, $c \in \mathbb{R}$, and code to reproduce the examples in this paper, cf. [23].

The control task is as follows. The goal is to track a known reference signal, $r_{0:T}$, with $T = 40$, cf. Fig. 1. The cost function minimized in the predictive control problem (9) is $\|\bar{y} - r\|_1$, where $\bar{y}$ denotes the predicted output with nominal disturbances $\bar{w} = 0$, cf. §III-D.4. The input and output constraints, $\mathcal{U}_t$ and $\mathcal{Y}_t$, are $|u_t| \leq 1$ and $y_t \leq 1.1$ for all t, respectively. The disturbances lie in the set $\mathcal{W} = \{w_t \in \mathbb{R} \mid |w_t| \leq 1\}$. In the initial data set $\mathcal{D}$, w^{d} constitute truncated Gaussian white noise; however, during control, a low-pass filter was applied (before truncation). The initial data set $\mathcal{D}$ is generated by simulating the system, open-loop, for $N = 500$ time-steps, excited by (non-truncated) unit variance Gaussian noise as input. We compare three methods: i) `closed-loop`: the closed-loop DeePC algorithm proposed in the present paper, ii) `open-loop`: an open-loop DeePC algorithm, similar to that proposed in [1], iii) `non-robust`: an open-loop DeePC algorithm, similar to [11], which does not take into consideration worst-case future disturbances; i.e., constraints are only enforced for the nominal predicted output $\bar{y}$. We emphasize that the only difference between `closed-loop` and `open-loop` is that the former includes decision variable $\mathcal{K}$, cf. (10), whereas the latter has $\mathcal{K} \equiv 0$. As such, any differences in performance can be attributed entirely to the presence/absence of disturbance feedback. A prediction horizon of $L = 10$ is selected, and policies/input sequences are redesigned every 5 time steps. All methods include a regularization term on (the approximate) solution g to (6) in the optimization objective, specifically, $\lambda \left\|M_\pi \nu + D^\perp z\right\|_2^2$, with $\lambda = 100$, though performance was qualitatively similar for $\lambda \in [5, 500]$ We set $T_i = 20$, though performance was similar for $T_i \in [4, 50]$.

Results are presented in Fig. 1 and Table I. The `closed-loop` policy achieves a 36% reduction (mean) in tracking error (as measured by the 1-norm) compared to the `open-loop` strategy. Although, the tracking error of `closed-loop` is 4% larger than that of `non-robust`, the latter violates the constraints in 90% of the experimental trials (as it does not account for worst-case future disturbances), while the proposed method recorded no constraint violations. The effect of feedback is transparent: after each policy redesign, the `open-loop` strategy tends to move away from the reference signal, when $r_t = 1$ (i.e. close to the constraint $y_t \leq 1.1$). This is necessary to ensure that the response to the open-loop input sequence does not violate the output constraint. In contrast, the `closed-loop` strategy can respond to disturbances after the policy redesign, enabling the system to remain closer to the constraint boundary without violating the constraint. We mention in passing

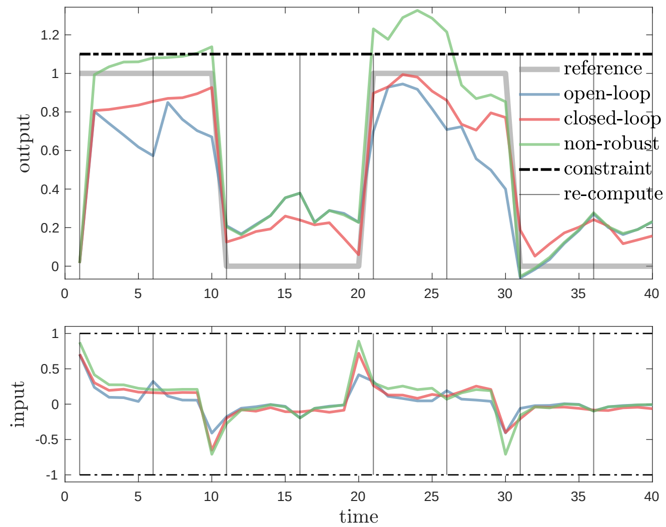

Fig. 1: Representative results from one of the trials in the numerical experiment outlined in §IV. The `closed-loop` controller tracks the reference signal more accurately than the `open-loop` controller, especially when the reference signal is close to the constraint boundary, while avoiding the constraint violations that occur with the `non-robust` controller.

TABLE I: Comparison of different controllers, as in §IV. Mean (std. dev.) quantities are reported for 10 trails. 'Violations' denotes the proportion of trials for which a constraint violation occurred. Programs parsed with [24], solved with Mosek, using an Intel i7 with 16GB of RAM.

Method	$\|y - r\|_1$	compute time (sec)	violations
`open-loop`	11.5 (1.79)	0.115 (0.007)	0%
`closed-loop`	7.35 (0.985)	0.169 (0.0253)	0%
`non-robust`	7.09 (1.33)	0.123 (0.006)	90%

that introducing a lower bound on the output, namely $y_t \geq y^{\min}$ for all t, renders the `open-loop` synthesis problem infeasible when $y^{\min} > -0.4$. Conversely, `closed-loop` synthesis remains feasible, even with $y^{\min} = 0$.

In closing, we remark that we do not consider this numerical example to be conclusive evidence of the superiority of the proposed approach, but rather an illustration of its principal merits, namely: reduced conservatism when constraints must be satisfied despite disturbances, in situations where the policy cannot be redesigned at every time-step. Furthermore, the illustration compares only DeePC methods. Rigorous benchmarking of data-enabled methods, such as DeePC, against more conventional "identification + MPC" approaches is an interesting and important direction for future research, but beyond the scope of this paper.

V. Conclusion

This paper aspires to deliver the following key message: if one wishes to apply the data-enabled predictive control method of [1], then superior performance can likely be achieved by employing the proposed closed-loop strategy. Specifically, the closed-loop formulation is guaranteed to return a solution at least as good as the open-loop formulation, for a modest increase in computational effort. Furthermore, the closed-loop method returns a policy that reacts to disturbances, as opposed to a fixed, open-loop sequence of inputs. Important and interesting directions for future work include extending the approach to systems for which disturbances are not directly measurable, as well as principled, data-enabled methods for selecting regularization parameters.

On the relationship to [1]

For the purpose of garnering feedback, a draft of this manuscript was shared with the authors of [1]. This correspondence revealed that the authors of [1] were working, entirely independently, on extending their method to incorporate disturbance feedback. The latest version of their preprint now briefly addresses disturbance feedback, cf. [25, §IV.B]. Nevertheless, we believe the two approaches are sufficiently different to warrant publication of the present manuscript. Specifically, while the basic idea of disturbance feedback appears in [25], none of the developments in §III-D are present.

Acknowledgements

The author would like to thank Prof. John Lygeros and Prof. Florian Dörfler for useful discussions and feedback on an early version of this manuscript. This work was supported by the National Science Foundation, Award No. EFMA-1830901, and the Department of Navy, Office of Naval Research, Award No. N00014-18-1-2210.

References

[1] L. Huang, J. Coulson, J. Lygeros, and F. Dörfler, "Data-driven wide-area control," *arXiv preprint arXiv:1911.12151v1*, 2019.

[2] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[3] R. E. Kalman *et al.*, "Contributions to the theory of optimal control," *Bol. soc. mat. mexicana*, vol. 5, no. 2, pp. 102–119, 1960.

[4] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall, 1999.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[6] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, p. 484, 2016.

[7] N. Matni, A. Proutiere, A. Rantzer, and S. Tu, "From self-tuning regulators to reinforcement learning and back again," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 3724–3740.

[8] S. Sadraddini and R. Tedrake, "Linear encodings for polytope containment problems," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 4367–4372.

[9] 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, 2005.

[10] I. Markovsky and P. Rapisarda, "Data-driven simulation and control," *International Journal of Control*, vol. 81, no. 12, pp. 1946–1959, 2008.

[11] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 307–312.

[12] ——, "Regularized and distributionally robust data-enabled predictive control," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2696–2701.

[13] J. Berberich, J. Köhler, M. A. Muller, and F. Allgower, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, 2020.

[14] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality and robustness," *IEEE Transactions on Automatic Control*, 2019.

[15] H. J. Van Waarde, J. Eising, H. L. Trentelman, and M. K. Camlibel, "Data informativity: a new perspective on data-driven analysis and control," *IEEE Transactions on Automatic Control*, 2020.

[16] P. J. Goulart, E. C. Kerrigan, and J. M. Maciejowski, "Optimization over state feedback policies for robust control with constraints," *Automatica*, vol. 42, no. 4, pp. 523–533, 2006.

[17] J. Lofberg, "Approximations of closed-loop minimax mpc," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, vol. 2. IEEE, 2003, pp. 1438–1442.

[18] S. J. Garstka and R. J.-B. Wets, "On decision rules in stochastic programming," *Mathematical Programming*, vol. 7, no. 1, pp. 117–143, 1974.

[19] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[20] G. E. Stewart, "A pragmatic approach to robust gain scheduling," *IFAC Proceedings Volumes*, vol. 45, no. 13, pp. 355–362, 2012.

[21] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[22] L. El Ghaoui, F. Oustry, and H. Lebret, "Robust solutions to uncertain semidefinite programs," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 33–52, 1998.

[23] J. Umenberger, "Closed-loop data-enabled control code." 2020. [Online]. Available: https://github.com/umenberger/robust-deepc

[24] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[25] L. Huang, J. Coulson, J. Lygeros, and F. Dörfler, "Decentralized data-enabled predictive control for power system oscillation damping," *arXiv preprint arXiv:1911.12151v3*, 2020.