Abstract—In this paper, we study a data-enabled predictive control (DeePC) algorithm applied to unknown stochastic linear time-invariant systems. The algorithm uses noise-corrupted input/output data to predict future trajectories and compute optimal control policies. To robustly against uncertainties in the input/output data, the control policies are computed to minimize a worst-case expectation of a given objective function. Using techniques from distributionally robust stochastic optimization, we prove that for certain objective functions, the worst-case optimization problem coincides with a regularized version of the DeePC algorithm. These results support the previously observed advantages of the regularized algorithm. We illustrate the robustness of the regularized algorithm through a numerical case study.

I. INTRODUCTION

Data-driven control approaches have become increasingly popular in recent years due to the growing complexity of systems [1]. The increase in system complexity poses significant challenges to model-based control design, in which acquiring an accurate model for the system is the most crucial step, and often the most time-consuming and expensive step [2]. The goal of data-driven control approaches is to bypass the system identification step, and directly use system measurements for control design.

The focus of this paper is on the problem of data-driven control for unknown systems. We seek to develop a data-driven control algorithm that uses input/output data from the unknown system to compute control inputs which satisfy system constraints. Furthermore, since uncertainties and noise corruption in the data measurements are inevitable in any real-world application, we seek algorithms that are robust to the uncertain and noisy data measurements used.

Several approaches such as safe-learning, learning based Model Predictive Control (MPC), and stochastic MPC aim at solving similar problems (see, e.g., [3]–[5] and references therein). However, these methods rely on having an a-priori known model for the system, require full-state measurement, and often assume knowledge about the disturbances that act on the system (e.g., bounded disturbances within a known set). Another large area of connected literature is reinforcement learning (see [6] and references therein). These methods also address similar problems, but usually require a large number of data measurements. Additionally, these approaches sometimes result in unreliable outcomes due to the sensitivity of the algorithms to hyper-parameters [7]. Hence, none of the approaches above are suitable for real-time, constrained, and optimal control based only on input/output samples.

We employ an alternative approach which does not rely on a particular parametric system model or assume any knowledge of the disturbances. The approach uses behavioural system theory to characterize the behaviour of the unknown system (i.e., the possible trajectories of the system) [8]. This behavioural approach was used for control design in [9] where optimal open-loop control policies were computed using input/output data from the unknown system. This was later extended in [10] to a receding horizon set up, which was proved to have equivalent closed-loop performance when compared to standard MPC in the case of deterministic linear time-invariant (LTI) systems.

The approach presented in this paper is built upon the Data-enabled Predictive Control (DeePC) algorithm for deterministic linear systems presented in [10]. The DeePC algorithm uses raw input/output data to compute optimal controls for unknown systems using real-time output feedback via a receding horizon implementation, thus allowing for the consideration of input/output constraints. In turn, this approach is much simpler to implement than the safe-learning, MPC, and reinforcement learning approaches above, which in the case of an unknown system with only output measurements available, require system identification and state observer design. On the other hand, the DeePC algorithm may not be amenable to stochastic systems, as the theory is built on the assumption of a deterministic LTI system. However, it was observed in [10] that after adding heuristic regularizations to the algorithm, DeePC still performed well on systems in the presence of stochastic disturbances, yet no robustness guarantees were given. We focus on extending this algorithm for use on stochastic systems. In doing so, we will give rigorous meaning to the heuristic regularizations presented in [10] by means of distributional robustness.

Due to the receding horizon implementation of the DeePC algorithm, we are required to repeatedly solve an optimization problem whose objective function naturally depends on data measurements that are affected by unknown disturbances. In order to be robust to these disturbances, we use distributionally robust optimization techniques [11], [12] which give rise to a regularized version of the DeePC algorithm similar to the heuristics presented in [10].

Contributions: Motivated by distributionally robust optimization and its connection to regularized optimization [11], [12], we develop a novel robust DeePC algorithm which uses input/output data to compute optimal and robust control inputs for unknown linear stochastic systems. By introducing distributional robustness, we give rigorous justification for
the use of the regularized control algorithm given in [10]. As a by-product, we gain new insights into the design of such regularizers.

Organization: Section II contains the problem statement. In Section III, we recall the DeePC algorithm and show how robust stochastic optimization techniques can be applied to improve the algorithm. Section IV contains the main result showing that regularizations on the DeePC algorithm result in robust performance. We illustrate the performance of the regularized algorithm in Section V. The paper is concluded in Section VI.

Notation: Given \( x, y \in \mathbb{R}^n \), \( \langle x, y \rangle := x^T y \) denotes the usual inner product on \( \mathbb{R}^n \times \mathbb{R}^n \). We denote the dual norm of a norm \( \| \cdot \| \) on \( \mathbb{R}^n \) by \( \| x \|_* := \sup_{\| y \| \leq 1} \langle x, y \rangle \). The conjugate function of a function \( f : \mathbb{R}^n \to \mathbb{R} \) is denoted by \( f^* (\theta) = \sup_{x \in \mathbb{R}^n} \langle \theta, x \rangle - f (x) \). We denote the stacked column vector of \( x \) and \( y \) by \( \text{col}(x,y) := (x^T, y^T)^T \). We denote by \( \delta_{\Delta} \) the Dirac distribution at \( x \). We use the notation \( \overset{\sim}{\cdot} \) to denote objects that depend on measured data.

II. Problem Statement

Consider the stochastic LTI system

\[
\begin{align*}
\dot{x}(t+1) &= Ax(t) + Bu(t) + Ev(t) \\
y(t) &= Cx(t) + Du(t) + Fv(t),
\end{align*}
\]

where \( A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{p \times n}, D \in \mathbb{R}^{p \times m}, E \in \mathbb{R}^{p \times q}, F \in \mathbb{R}^{p \times q}, \) and \( x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m, y(t) \in \mathbb{R}^p, v(t) \in \mathbb{R}^q \) respectively the state, control input, output, and stochastic disturbance of the system at time \( t \in \mathbb{Z}_{\geq 0} \). The disturbance \( v(t) \) is drawn from an unknown probability distribution \( \mathbb{P}_v \) for all \( t \in \mathbb{Z}_{\geq 0} \). We assume throughout the paper that the pair \((A,B)\) is controllable, the pair \((A,C)\) is observable, that system \((1)\) itself is unknown (i.e., \(A,B,C,D,E,F\) unknown), and that we have access only to input/output measurements, which we will denote by \( \hat{u}(t), \hat{y}(t) \).

We consider a finite-horizon optimal control problem, in which the task is to design control inputs for the unknown system to minimize a given objective function. More specifically, given a time horizon \( T \in \mathbb{Z}_{>0} \), an objective function \( f : \mathbb{R}^{mT} \times \mathbb{R}^{pT} \to \mathbb{R} \), and the current time \( t \in \mathbb{Z}_{\geq 0} \), we wish to choose a sequence of control inputs \( u = \text{col}(u(t), \ldots, u(T + T_i - 1)) \in \mathcal{U} \), where \( \mathcal{U} \subset \mathbb{R}^{mT} \) is an input constraint set, such that the resulting stochastic output trajectory, \( y = \text{col}(y(t), \ldots, y(T + T_i - 1)) \), of (1) minimizes the expectation of the objective function, i.e., we wish to solve the following optimization problem:

\[
\inf_{u \in \mathcal{U}} \mathbb{E}_{\mathcal{P}_v^T} \{ f(u, y) \},
\]

where \( \mathcal{P}_v^T = \mathcal{P}_v \times \cdots \times \mathcal{P}_v \) is the \( T \)-fold product distribution. Such optimization problems also appear in stochastic MPC [13].

Solving problem (2) poses two main challenges: the system (1) is unknown, and the disturbance distribution \( \mathbb{P}_v \) is unknown. Hence, we are unable to predict future output trajectories \( y \), and we are missing necessary information to compute the expectation. Note that even in the case when system (1) and \( \mathbb{P}_v \) are known, solving (2) would require high-dimensional integration and is often computationally intractable [14]. One could also consider including joint output chance constraints in the problem setup (2) which would pose similar challenges as above. We do not consider such constraints as they are beyond the scope of this paper.

To address these challenges, we first simplify the problem by considering the deterministic version of problem (1):

\[
\begin{align*}
x(t+1) &= Ax(t) + Bu(t) \\
y(t) &= Cx(t) + Du(t),
\end{align*}
\]

In this case, problem (2) reduces to

\[
\inf_{u \in \mathcal{U}} f(u, y).
\]

We recall a data-enabled predictive control (DeePC) algorithm first presented in [10] that can be used to solve (4) for deterministic though unknown systems (3) (see Section III-A). We then show how this algorithm can be extended to approach problem (2) for stochastic systems (1) using distributionally robust optimization techniques (see Section III-B). The main result can be found in Section IV.

III. Preliminaries

A. DeePC Algorithm for Deterministic Systems

In this section, we recall the DeePC algorithm which uses raw input/output data to construct a non-parametric predictive model that was first developed using a behavioural system theory approach in [8]. Let \( L, q, T \in \mathbb{Z}_{>0} \). We define the Hankel matrix of a signal \( w \in \mathbb{R}^{pT} \) with \( L \) block rows as the matrix

\[
\mathcal{H}_L(w) := \begin{pmatrix}
w(1) & w(2) & \cdots & w(T - L + 1) \\
w(2) & w(3) & \cdots & w(T - L + 2) \\
\vdots & \vdots & \ddots & \vdots \\
w(L) & w(L+1) & \cdots & w(T)
\end{pmatrix}.
\]

We say that signal \( w \) is persistently exciting of order \( L \) if \( \mathcal{H}_L(w) \) has full row rank. Note that in order for \( w \) to be persistently exciting of order \( L \), we must have \( T \geq (q + 1)L - 1 \), i.e., the signal \( w \) has to be sufficiently rich and sufficiently long. The lag of system (3) is defined as the smallest integer \( \ell \in \mathbb{Z}_{>0} \) such that the observability matrix \( \Omega_{\ell}(A, C) := \text{col}(C, CA, \ldots, CA^{\ell-1}) \) has rank \( n \) (see [15, Section 7.2] for equivalent state space free definitions of lag).

Like in Section II, we assume that we have access to input/output measurements. Let \( \text{col}(\hat{u}(1), \hat{y}(1), \hat{u}(T), \hat{y}(T)) \in \mathbb{R}^{(m+p)L} \) be a measured trajectory of (3) of length \( T \in \mathbb{Z}_{>0} \). Assume that \( \hat{u} \) is persistently exciting of order \( T_{\text{in}} + T_{\ell} + n \), where \( T_{\text{in}}, T_{\ell} \in \mathbb{Z}_{>0} \). We organize the data into the Hankel matrices

\[
\begin{align*}
\hat{U}_{\text{in}} &:= \mathcal{H}_{T_{\text{in}} + T_{\ell}}(\hat{u}), \\
\hat{Y}_{\text{in}} &:= \mathcal{H}_{T_{\text{in}} + T_{\ell}}(\hat{y}),
\end{align*}
\]

where \( \hat{U}_{\text{in}} \) consists of the first \( T_{\text{in}} \) block rows of \( \mathcal{H}_{T_{\text{in}} + T_{\ell}}(\hat{u}) \) and \( \hat{U}_{\text{ell}} \) consists of the last \( T_{\ell} \) block rows of \( \mathcal{H}_{T_{\text{in}} + T_{\ell}}(\hat{u}) \).
Given the current time $t \in \mathbb{Z}_{\geq 0}$, let $\hat{u}_{\text{ini}} = \text{col}(\hat{u}(t - T_{\text{ini}}), \ldots, \hat{u}(t - 1)) \in \mathbb{R}^{mT_{\text{ini}}}$ and $\hat{y}_{\text{ini}} = \text{col}(\hat{y}(t - T_{\text{ini}}), \ldots, \hat{y}(t - 1)) \in \mathbb{R}^{pT_{\text{ini}}}$ be the most recent input and output measurements of (3), respectively. By [8, Theorem 1], $\text{col}(\hat{u}_{\text{ini}}, u, \hat{y}_{\text{ini}}, y) \in \mathbb{R}^{(m+p)(T_{\text{ini}}+T_{f})}$ is a trajectory of system (3) if and only if there exists $g \in \mathbb{R}^{T - T_{\text{ini}} - T_{f} + 1}$ such that

$$
\begin{pmatrix}
\hat{U}_p \\
\hat{Y}_p \\
\hat{U}_f
\end{pmatrix}
= 
\begin{pmatrix}
\hat{u}_{\text{ini}} \\
u \\
y
\end{pmatrix},
$$

(6)

Furthermore, if $T_{\text{ini}} \geq \ell$, then by [9, Lemma 1], there exists a unique $x(t) \in \mathbb{R}^n$ such that the output trajectory $y$ is uniquely determined by the system (3). In other words, the trajectory col($\hat{u}_{\text{ini}}, \hat{y}_{\text{ini}})$ fixes the underlying initial state $x(t)$ from which the trajectory $\text{col}(u, y)$ evolves. Hence, the Hankel matrix in (6) serves as a non-parametric predictive model for system (3). This observation was first exploited for control purposes in [9], where equation (6) was used to construct open-loop control policies for tracking. This was then extended in [10], where equation (6) was used in a receding horizon optimal control algorithm. We recall the latter approach below.

Given a time horizon $T_f \in \mathbb{Z}_{>0}$, past input/output data $\text{col}(\hat{u}_{\text{ini}}, \hat{y}_{\text{ini}}) \in \mathbb{R}^{(m+p)T_{\text{ini}}}$, objective function $f: \mathbb{R}^{mT_f} \times \mathbb{R}^{pT_f} \rightarrow \mathbb{R}$, we formulate the following optimization problem:

minimize $g$

subject to

$$
\begin{pmatrix}
\hat{U}_p \\
\hat{Y}_p \\
\hat{U}_f
\end{pmatrix}
= 
\begin{pmatrix}
\hat{u}_{\text{ini}} \\
u \\
y
\end{pmatrix},
$$

(7)

$\hat{U}_fg \in \mathcal{U}$.

Note that optimization problem (7) is equivalent to (4) since by (6), $u = \hat{U}_fg$ and $y = \hat{Y}_fg$. The data-enabled predictive control (DecPC) algorithm then solves the optimization problem (7) for $g^*$ and applies the first $s \leq T_f - 1$ steps of the optimal control input $u^* := \hat{U}_fg^*$, where $s$ is a user-defined parameter. The measurements $\hat{u}_{\text{ini}}$ and $\hat{y}_{\text{ini}}$ are then updated to the most recent input/output measurements and this process is repeated.

This algorithm was shown to be equivalent to the classical Model Predictive Control (MPC) algorithm when considering deterministic LTI systems of the form (3), also in the presence of additional output constraints. When the DeePC algorithm was applied to a nonlinear stochastic quadcopter model in [10], additional heuristic regularization terms had to be included in (7) to achieve robust performance. We recall the regularizations below.

minimize $g$

subject to

$$
\begin{aligned}
\hat{U}_fg &= \hat{u}_{\text{ini}} \\
\hat{U}_fg &\in \mathcal{U},
\end{aligned}
$$

(8)

where $\lambda_u, \lambda_y \in \mathbb{R}_{>0}$. We will show that this regularized optimization problem coincides with solving a distributionally robust variant of optimization problem (2), giving rigorous justification for the regularizations (see Section IV).

B. Distributionally Robust DeePC

Consider now the stochastic system (1). Let $\text{col}(\hat{u}, \hat{y}) \in \mathbb{R}^{(m+p)T}$ be a measured trajectory of length $T$ of system (1) such that $\hat{u}$ is persistently exciting of order $T_{\text{ini}} + T_{f} + n$, where $T_{\text{ini}} \geq \ell$. Furthermore, assume $\text{col}(\hat{u}_{\text{ini}}, \hat{y}_{\text{ini}}) \in \mathbb{R}^{(m+p)T_{\text{ini}}}$ is the last measured trajectory of system (1). The output trajectories $\hat{y}$ and $\hat{y}_{\text{ini}}$ can be viewed as particular realizations of random variables, $y$ and $y_{\text{ini}}$. If we naively use the noise corrupted trajectory $\hat{y}$ to build Hankel matrices $\hat{U}_p, \hat{Y}_p, \hat{U}_f, \hat{Y}_f$ as in (5) and apply the DeePC algorithm, we will run into difficulties regarding the consistency of the constraint equations in (7). Indeed, since $\hat{y}_{\text{ini}}$ and $\hat{Y}_p$ consist of noise corrupted outputs, there may not exist $g$ that satisfies the equation $\hat{Y}_pg = \hat{y}_{\text{ini}}$. Hence, we soften the equality constraint and penalize the slack variable with an appropriate cost function; we recall that the use of such “soft constraints” is common in MPC [16]. This results in the optimization problem

minimize $g$

subject to

$$
\begin{aligned}
\hat{U}_fg &= \hat{u}_{\text{ini}} \\
\hat{U}_fg &\in \mathcal{U},
\end{aligned}
$$

(9)

where we have used the 1-norm as a penalty function on the slack variable. It is well known that if $\lambda_{\text{ini}}$ is chosen large enough then the solution $g^*$ will violate $\hat{Y}_pg^* = \hat{y}_{\text{ini}}$ only if the equation is infeasible [17]. Hence, the original constraint will be satisfied if it can be satisfied. Since the constraint is now deterministic, we may define $G = \{g \in \mathbb{R}^{T - T_{\text{ini}} - T_{f} + 1} | \hat{U}_fg = \hat{u}_{\text{ini}}, \hat{U}_fg \in \mathcal{U}\}$ and rewrite the above as

minimize $g \in G$

subject to

$$
\begin{pmatrix}
\hat{U}_p \\
\hat{Y}_p
\end{pmatrix}
= \begin{pmatrix}
\hat{u}_{\text{ini}} \\
y_{\text{ini}}
\end{pmatrix},
$$

(10)

where $\{\cdot\}_j$ denotes the $j$-th row of a matrix. For all $j \in \{1, \ldots, p(T_{\text{ini}} + T_{f})\}$, let $P_j$ denote the probability distribution of $\xi_j$ supported on $\Xi_j \subseteq \mathbb{R}^{T - T_{\text{ini}} - T_{f} + 2}$. Note that the distributions $P_j$ and their support sets $\Xi_j$ are determined by the dynamics of system (1) and the unknown distribution $P_v$.

Define $\xi = \text{col}(\xi_1^T, \ldots, \xi_p^T(T_{\text{ini}} + T_{f}))$ and let $P := \mathbb{P}_1 \times \cdots \times \mathbb{P}_{p(T_{\text{ini}} + T_{f})}$ denote the unknown probability distribution of $\xi$ supported on $\Xi = \prod_{j=1}^{p(T_{\text{ini}} + T_{f})} \Xi_j \subseteq \mathbb{R}^{p(T_{\text{ini}} + T_{f})(T - T_{\text{ini}} - T_{f} + 2)}$. Let $H = \{h = \text{col}(g, -1) | \hat{U}_fg = \hat{u}_{\text{ini}}, \hat{U}_fg \in \mathcal{U}\}$. Substituting this notation into (9) yields

minimize $h \in H$

subject to

$$
\begin{pmatrix}
\hat{U}_1, 0 \\
\xi_1, \ldots, \xi_p(T_{\text{ini}} + T_{f})
\end{pmatrix}
= \begin{pmatrix}
\hat{u}_{\text{ini}} \\
\hat{Y}_p
\end{pmatrix},
$$

(11)

+ $\lambda_{\text{ini}}(\|\xi_1 - h_1 - \ldots - \xi_p(T_{\text{ini}} + T_{f})\|_1)$,
where \( \hat{\xi} = \text{col}(\hat{\xi}_1, \ldots, \hat{\xi}_p(T_{\text{ini}}+T_f)) \) are our measurements. Denoting the empirical distribution of our measurements \( \hat{\xi} \) by \( \hat{P} = \delta_{\hat{\xi}} \), and substituting this notation into the above yields

\[
\min_{h \in \mathcal{H}} \mathbb{E}_{\hat{P}} \{ c(\xi, h) \},
\]

where \( c(\xi, h) = f((\hat{U}_1, 0)h, (\xi_{pT_{\text{ini}}+1}h, \ldots, \xi_{p(T_{\text{ini}}+T_f)}h)) + \lambda_{\text{lin}} \| (\xi_1h, \ldots, \xi_p(T_{\text{ini}}+T_f)h) \|_1. \)

The quantity in (10) is known as the in-sample performance. Unfortunately, the solution \( h^* \) to (10) may result in poor out-of-sample performance

\[
\mathbb{E}_{\hat{P}} \{ c(\xi, h^*) \},
\]

which if we compare to equation (2) is the real quantity of interest (see, e.g., [11], [18] for examples displaying poor out-of-sample performance).

To alleviate this problem, we focus instead on a robust variation of (10). In particular, we seek solutions of

\[
\min_{h \in \mathcal{H}} \sup_{Q \in \mathcal{P}} \mathbb{E}_Q \{ c(\xi, h) \},
\]

where \( \mathcal{P} \) is an ambiguity set which depends on the sampled data trajectories \( \hat{\xi} \).

Motivated by the strong out-of-sample performance of distributionally robust optimization problems under Wasserstein ambiguity sets in [11], we define the ambiguity set in (12) in terms of the Wasserstein metric defined on the space \( \mathcal{M}(\Xi) \) denoting the set of all distributions \( Q \) supported on \( \Xi \) such that \( \mathbb{E}_Q \| \xi \|_W < \infty \), where \( \| \cdot \|_W \) is an arbitrary norm.

**Definition 3.1:** The Wasserstein metric \( d_W : \mathcal{M}(\Xi) \times \mathcal{M}(\Xi) \to \mathbb{R}_{\geq 0} \) is defined as

\[
d_W(Q_1, Q_2) := \inf_{\Pi} \left\{ \mathbb{E}_{\Xi^2} [\|\xi_1 - \xi_2\|_W^2] \right\},
\]

where \( \Pi \) is a joint distribution of \( \xi_1 \) and \( \xi_2 \) with marginal distributions \( Q_1 \in \mathcal{M}(\Xi) \) and \( Q_2 \in \mathcal{M}(\Xi) \) respectively.

Let \( \epsilon \geq 0 \). We denote the Wasserstein ball of radius \( \epsilon \) centred around distribution \( Q \) by \( B_\epsilon(Q) := \{ Q' \in \mathcal{M}(\Xi) \mid d_W(Q, Q') \leq \epsilon \} \). The Wasserstein metric can be viewed as a distance between probability distributions, where the distance is calculated via an optimal mass transport plan \( \Pi \). Note that there are other ways to construct ambiguity sets (see, e.g., [18] where the ambiguity set is constructed as the confidence region of a goodness-of-fit hypothesis test). Replacing the general ambiguity set \( \mathcal{P} \) with a Wasserstein ball around the empirical distribution \( \hat{P} \) results in the problem

\[
\min_{h \in \mathcal{H}} \sup_{Q \in B_\epsilon(\hat{P})} \mathbb{E}_Q \{ c(\xi, h) \}.
\]

In the next section, we show that (13) is computationally tractable and results in a regularized version of the DeePC algorithm similar to (8).

### IV. Main Result

By adapting the proof methods of [11] and [12] to our setting, we show that for certain objective functions, the semi-infinite optimization problem (13) reduces to a tractable convex program that coincides with a regularized version of (10).

**Theorem 4.1:** (Tractable Reformulation): Let \( \| \cdot \| \) be an arbitrary norm on \( \mathbb{R}^{T-T_{\text{ini}}-T_f+2} \) and \( \| \xi \|_W = \sum_{j=1}^{p(T_{\text{ini}}+T_f)} \| \xi_j \| \) be the norm used in the Wasserstein metric. Assume that the objective function \( f \) is separable and can be written as \( f(u, y) = f_1(u) + f_2(y) \) for all \( u, y \), where \( f_1 \) and \( f_2 \) are convex and continuous. Furthermore, assume \( f_2 \) is such that 

\[
\Theta_2 = \{ \theta \mid f_2(\theta) < \infty \}
\]

is a bounded set in \( \mathbb{R}^{T_f} \). Then

\[
\sup_{Q \in B_\epsilon(\hat{P})} \mathbb{E}_Q \{ c(\xi, h) \}
\]

\[
\leq c(\hat{\xi}, h) + \epsilon \max \left\{ \sup_{\theta \in \Theta_2} \| \theta \|_W \| \text{col}(g, 0) \|_* , \lambda_{\text{lin}} \| h \|_* \right\},
\]

where \( \| \cdot \|_* \) denotes the dual norm of \( \| \cdot \| \). Equality holds when \( \Xi_j = \mathbb{R}^{T-T_{\text{ini}}-T_f+2} \) for all \( j \in \{ 1, \ldots, pT_{\text{ini}} \} \) and \( \Xi_j = \mathbb{R}^{T-T_{\text{ini}}-T_f+1} \times \{ 0 \} \) for all \( j \in \{ pT_{\text{ini}}+1, \ldots, p(T_{\text{ini}}+T_f) \} \).

This theorem gives rise to the robust DeePC algorithm.

**Algorithm 1 Robust DeePC**

**Input:** trajectory \( \text{col}(\hat{u}, \hat{y}) \in \mathbb{R}^{(m+p)T} \) with \( \hat{u} \) persistently exciting of order \( T_{\text{ini}} + T_f + n \), most recent input/output measurements \( \text{col}(\hat{u}_{\text{ini}}, \hat{y}_{\text{ini}}) \in \mathbb{R}^{(m+p)T_{\text{ini}}} \)

1. Set \( h^* \) equal to

\[
\arg \min_{h \in \mathcal{H}} \left\{ c(\hat{\xi}, h)\right. \\
+ \epsilon \max \left\{ \sup_{\theta \in \Theta_2} \| \theta \|_W \| \text{col}(g, 0) \|_* , \lambda_{\text{lin}} \| h \|_* \right\} \right\}.
\]

2. Compute the optimal input sequence \( u^* = \hat{U}g^* \), where \( h^* = \text{col}(g^*, -1) \).

3. Apply input \( (u(t), \ldots, u(t+s)) = (u_0^*, \ldots, u_s^*) \) for some \( s \leq T_1 - 1 \).

4. Set \( t \to t+s \) and update \( \hat{u}_{\text{ini}} \) and \( \hat{y}_{\text{ini}} \) to the \( T_{\text{ini}} \) most recent input/output measurements.

5. Return to 1.

Note that the condition for equality holds when the disturbance \( v \) affecting system (1) is drawn from a distribution \( P_v \) defined on an unbounded support set and the matrix \( F \) in (1) is full row rank. In other words, equality will hold if the probability distribution \( P_v \) has unbounded support and disturbances \( v(t) \) affect all entries of the output vector \( y(t) \). In this case, each probability distribution \( P_j \) would have support \( \Xi_j = \mathbb{R}^{T-T_{\text{ini}}-T_f+2} \) for \( j \in \{ 1, \ldots, pT_{\text{ini}} \} \) and \( \Xi_j = \mathbb{R}^{T-T_{\text{ini}}-T_f+1} \times \{ 0 \} \) for \( j \in \{ pT_{\text{ini}}+1, \ldots, p(T_{\text{ini}}+T_f) \} \). For example, we would obtain equality in the above if \( F \) is full row rank and \( P_v \) is a Gaussian distribution which is a common assumption in stochastic MPC (see [5] and references therein).
We see an immediate connection between Theorem 4.1 and (8). When \( \| \cdot \|_* = \| \cdot \|_1 \) then \( \epsilon\max_{\theta \in \Theta_2} \| \theta \|_\infty \| g \|_1, \lambda_{\text{ini}} \| (g_1 + 1) \|_\star \) = \( \epsilon\max_{\theta \in \Theta_2} \| \theta \|_\infty \| g \|_1, \lambda_{\text{ini}} \| (g_1 + 1) \|_\star \). Hence, depending on the known cost function \( f_2 \) (i.e., on the set \( \Theta_2 \)), \( \lambda \) in (8) plays the role of either \( \epsilon\sup_{\theta \in \Theta_2} \| \theta \|_\infty \) or \( \epsilon\lambda_{\text{ini}} \). Thus, the result of Theorem 4.1 gives insight into the design of the regularizer on the decision variable \( g \) to achieve various robustness goals for DEPC.

Remark 4.1: The assumption that \( \Theta_2 = \{ \theta \mid f_2(\theta) < \infty \} \) is a bounded set in \( \mathbb{R}^{pT} \) holds for many objective functions of practical interest. Indeed, being robust in the trajectory space in the \( \| \cdot \|_\infty \) sense requires regularizing with a \( \| \cdot \|_1 \). Likewise, being robust in the trajectory space in the \( \| \cdot \|_2 \) sense requires regularizing with a \( \| \cdot \|_2 \). If no noise is present in the system, we may set \( \epsilon = 0 \) recovering the DEPC algorithm for deterministic LTI systems.

Remark 4.2: The norm \( \sum_{j=1}^{p(T_n+1)} \| \xi_j \| \) used in the Wasserstein metric can be chosen to achieve various robustness goals. Indeed, being robust in the trajectory space in the \( \| \cdot \|_\infty \) sense requires regularizing with a \( \| \cdot \|_1 \). Likewise, being robust in the trajectory space in the \( \| \cdot \|_2 \) sense requires regularizing with a \( \| \cdot \|_2 \). If no noise is present in the system, we may set \( \epsilon = 0 \) recovering the DEPC algorithm for deterministic LTI systems.

Remark 4.3: The application of Algorithm 1 is not limited only to LTI systems with additive noise. This is due to the fact that we are solving a worst-case optimization problem over a Wasserstein ball containing probability distributions that could have given rise to the measured trajectories. The distributions within this ball may also be capable of giving rise to trajectories from more complex systems, e.g., nonlinear systems, or systems with multiplicative noise.

To prove the theorem above, we require the following lemma which is an extension of [12, Lemma A.3] to functions with vector valued inputs. The proof can be found in [19].

Lemma 4.1: Let \( a, b, \alpha, \beta \in \mathbb{Z}_{\geq 0} \), and \( \Omega_j \subseteq \mathbb{R}^{q} \) for \( j \in \{1, \ldots, r\} \). Let \( \xi_j \in \Omega_j \) be given for \( j \in \{1, \ldots, r\} \), \( \lambda \in \mathbb{R}_{\geq 0} \), and \( \alpha \in \mathbb{R} \) convex and continuous such that \( \{ \xi \mid L^\alpha(\theta) < \infty \} \) is a bounded set in \( \mathbb{R}^n \). Then, for fixed \( b \in \mathbb{R}^n \),

\[
\sup_{\xi_j \in \Omega_j} L(\xi_j^T b, \ldots, \xi_j^T b) - \lambda \sum_{j=1}^{r} \| \xi_j - \hat{\xi}_j \|_\infty \lesssim \|
\leq \sum_{j=1}^{r} \| \xi_j - \hat{\xi}_j \|_\infty \lesssim \|
\]

where \( \| \cdot \| \) is an arbitrary norm on \( \mathbb{R}^n \). Furthermore, the above is an equality when \( \Omega_j = \mathbb{R}^n \) for all \( j \in \{1, \ldots, r\} \).

Proof: (Theorem 4.1): Since \( c(\cdot, h) \) is a proper, convex, continuous function for all \( h \in H \) then

\[
\sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \} = \sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \}
\]

This can be shown by studying the dual problem, and noticing that the worst case distribution coincides with a Dirac distribution at the point \( \xi \) which results in the largest norm \( \| \xi - \hat{\xi} \|_w \) (see [11, Theorem 6.3]). Define \( f_3(\cdot) := \lambda_{\text{ini}} \| \cdot \|_1 \). By separability of the objective function,

\[
\sup_{Q \in B_{\scriptscriptstyle \mathcal{B}}(\bar{\rho})} \mathbb{E}_Q \{ c(\xi, h) \} \leq \sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \} = \sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \}
\]

with equality when \( \Xi_j = \mathbb{R}^{T-T_n-\tilde{r}+2} \), \( \Xi_j = \mathbb{R}^{T-T_n-\tilde{r}+2} \times \{0\} \) for all \( j \in \{pT_n+1, \ldots, p(T_n+1)\} \). By reversing the epigraph formulation and the definition of \( \| \cdot \|_w \) for the last equality. By definition of the conjugate function, \( f_3^*(\theta) = 0 \) if \( \| \theta \|_\infty \leq \lambda_{\text{ini}} \) and is infinite otherwise. Hence, \( \Theta_3 := \{ \theta \mid f_3^*(\theta) < \infty \} \) is a bounded set in \( \mathbb{R}^{pT} \). We also have that \( \Theta_2 := \{ \theta \mid f_2^*(\theta) < \infty \} \) by assumption. Hence, by Lemma 4.1,

\[
\sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \} = \sup_{\xi \in \Xi} \mathbb{E}_Q \{ c(\xi, h) \}
\]

Note that the infinite case is dropped since we are taking the
infimum over $\lambda \geq 0$. Substituting notation yields

$$
\sup_{Q \in B_{\epsilon}(\hat{P})} \mathbb{E}_Q \{c(\xi, h)\} \\
\leq c(\hat{\xi}, h) + \epsilon \cdot \max \left\{ \sup_{\theta \in \Theta} \|\theta\|_\infty, \lambda_{\text{ini}}, \|h\|_\infty \right\}.
$$

Noting that $\hat{h} = \text{col}(\theta, 0)$ gives the claimed result.

V. SIMULATIONS

We present results of the robust algorithm for various Wasserstein radii on a linear model of a quadcopter commanded to follow a step trajectory denoted by $r$. The states of the quadcopter model are given by the 3 spatial coordinates $(x, y, z)$ and their velocities, and the 3 angular coordinates $(\alpha, \beta, \gamma)$ and their velocities, i.e., the state is $(x, y, z, \dot{x}, \dot{y}, \dot{z}, \alpha, \beta, \gamma, \dot{\alpha}, \dot{\beta}, \dot{\gamma})$. The inputs are given by the individual thrusts from the 4 rotors. We assume full state measurement. The exact model can be found in [10].

By observing the performance of the quadcopter, we are able to estimate a range of optimal Wasserstein radii.

The following parameters were chosen for the optimization problem (13): 214 input/output measurements were used to populate the matrices $\tilde{U}_p$, $\tilde{Y}_p$, $\tilde{U}_t$, and $\tilde{Y}_t$, where the input was drawn from a uniform random variable to ensure persistency of excitation, the infinity-norm was used in the definition of the Wasserstein metric, $T_t = 30$, $T_{\text{ini}} = 7$, $f(u, y) = 0.1\|u\|_1 + 2000\|y - r\|_1$, $\lambda_{\text{ini}} = 10^5$, $\mathcal{U} = [-0.7007, 0.2993]^2$. The states were affected by zero-mean truncated Gaussian noise, where the truncation was made 3 standard deviations from both sides of the mean. The output measurements were also corrupted by a zero-mean truncated Gaussian distribution. A Wasserstein radius $\epsilon$ was fixed and the cost $f(u, y)$ accumulated over the 20 second horizon of the simulation was computed. The cost was averaged over 15 random simulations for each $\epsilon$.

The results shown in Figure 1 indicate that $\epsilon$ should be chosen in the interval $[0.001, 0.01]$. The results also support the claim that optimizing the in-sample performance (10) (i.e., setting $\epsilon = 0$) displays poor out-of-sample performance, hence justifying the robust approach. It is also clear from Figure 1 that choosing $\epsilon$ large (i.e., being over-conservative) results in poor performance.

VI. CONCLUSION

We studied the problem of controlling an unknown stochastic system. With no knowledge of the nature of the stochasticity, we proposed a robust DeePC algorithm which uses noise corrupted input/output data to compute optimal and robust control inputs. Robustifying the original DeePC algorithm gave rise to principled regularization terms, supporting the observed superior performance of the regularized algorithm on stochastic systems. Future work includes incorporating multiple measured data sets to improve performance, and including output constraints.

REFERENCES